Robust preconditioners via generalized eigenproblems for hybrid sparse linear solvers

Emmanuel Agullo, Luc Giraud, Louis Poirel

ROBUST PRECONDITIONERS VIA GENERALIZED EIGENPROBLEMS FOR HYBRID SPARSE LINEAR SOLVERS

E. AGULLO, L. GIRAUD, L. POIREL

Abstract. The solution of large sparse linear systems is one of the most time consuming kernels in many numerical simulations. The domain decomposition community has developed many efficient and robust methods in the last decades. While many of these solvers fall into the abstract Schwarz (aS) framework, their robustness has originally been demonstrated on a case-by-case basis. In this paper, we propose a bound for the condition number of all deflated aS methods provided that the coarse grid consists of the assembly of local components that contain the kernel of some local operators. We show that classical results from the literature on particular instances of aS methods can be retrieved from this bound. We then show that such a coarse grid correction can be explicitly obtained algebraically via generalized eigenproblems, leading to a condition number independent of the number of domains. This result can be readily applied to retrieve or improve the bounds previously obtained via generalized eigenproblems in the particular cases of Neumann-Neumann (NN), Additive Schwarz (AS) and optimized Robin but also generalizes them when applied with approximate local solvers. Interestingly, the proposed methodology turns out to be a comparison of the considered particular aS method with generalized versions of both NN and AS for tackling the lower and upper part of the spectrum, respectively. We furthermore show that the application of the considered grid corrections in an additive fashion is robust in the AS case although it is not robust for aS methods in general. In particular, the proposed framework allows for ensuring the robustness of the AS method applied on the Schur complement (AS/S), either with deflation or additively, and with the freedom of relying on an approximate local Schur complement. Numerical experiments illustrate these statements.

Key words. preconditioning, SPD linear systems, robust, scalable, coarse space, generalized eigenvalue, parallel hybrid (direct/iterative) solver

AMS subject classifications. 15A06, 65F08, 65F10, 15A12, 65N55

1. Introduction.

Many scientific or engineering applications require at some point the solution of large sparse linear systems in parallel. Once the specific problem has been discretized, the resulting matrix equation can be solved using either an external general purpose linear solver, or a more specific solver tailored to the particular problem. With the first approach, referred to as the algebraic approach, the user can benefit with little integration effort from the developments and optimizations of black-box libraries which perform very well on modern architectures [4, 19].

On the other hand, the second approach often allows additional optimizations that further exploit additional characteristics of the underlying problem and requires a tighter integration of the solver within the application code. A widely used class of methods that fall in this latter category are domain decomposition methods (DDM) [10, 20, 28, 30, 35], which are inherently parallel and provide robust and scalable solvers for a wide range of physical problems.

In this article, we aim at combining the advantages of both these approaches. For that, while remaining as algebraic as possible, we identify some key information to be provided to the solver alongside the matrix. For symmetric positive definite (SPD) problems we show that providing the matrix in a distributed fashion, as a sum of symmetric positive semi-definite (SPSD) matrices, is enough to build a robust and scalable hybrid solver. This is a common situation when applying a finite element method over a partitioned mesh, but the methods presented in this article are not limited to this particular case: for instance, more complex discretizations such as the hybridizable discountinuous Galerkin method [7] can be used instead.

The linear system to be solved is

$$\mathcal{K}u = f, \tag{1.1}$$

where $\mathcal{K}$ is a $n \times n$ sparse SPD matrix that does not need to be known explicitly. Instead, the parallel application provides $\mathcal{K}$ to the solver as a sum $\mathcal{K} = \sum_{i=1}^{N} \mathcal{K}_i^{(g)}$ of N SPSD matrices $\mathcal{K}_i^{(g)}$. Even though $\mathcal{K}_i^{(g)}$ is of size $n \times n$, in practical applications it has only n_i non-zero rows (and columns), meaning that this matrix represents the interaction of only a subset of the unknowns from the global problem. We define the *global domain* $\Omega = \{1, \ldots, n\}$ as the set of row (or column) indices in $\mathcal{K}$, and the *subdomain* $\Omega_i = \{\omega_1^{(i)}, \omega_2^{(i)}, \ldots, \omega_{n_i}^{(i)}\}$ as the set of indices of the non-zero rows and columns in $\mathcal{K}_i^{(g)}$ (Ω_i is the set of vertices in the adjacency graph of $\mathcal{K}_i^{(g)}$). We introduce the $n_i \times n$ canonical restriction matrix $\mathcal{R}_{\Omega_i}$ from Ω to Ω_i, such that for any vector $u = (u_1, \ldots, u_n) \in \mathbb{R}^n$, $\mathcal{R}_{\Omega_i} u$ is the vector $(u_{\omega_1^{(i)}}, \ldots, u_{\omega_{n_i}^{(i)}}) \in \mathbb{R}^{n_i}$. Then, we define the $n_i \times n_i$ SPSD matrix $\mathcal{K}_i = \mathcal{R}_{\Omega_i} \mathcal{K}_i^{(g)} \mathcal{R}_{\Omega_i}^T$, referred to as the *local matrix* of subdomain Ω_i, leading to

$$\mathcal{K} = \sum_{i=1}^{N} \mathcal{R}_{\Omega_i}^T \, \mathcal{K}_i \, \mathcal{R}_{\Omega_i}. \tag{1.2}$$

The unknowns in any subdomain Ω_i can be partitioned into an interior $\mathcal{I}_i = \{\omega \in \Omega_i \quad s.t. \quad \forall j \neq i, \quad \omega \notin \Omega_j\}$ and an interface $\Gamma_i = \{\omega \in \Omega_i \quad s.t. \quad \exists j \neq i \quad \omega \in \Omega_j\} = \Omega_i \setminus \mathcal{I}_i$. If an unknown $\omega \in \Omega_i$ appears in at least one other subdomain, then $\omega \in \Gamma_i$, otherwise $\omega \in \mathcal{I}_i$. This yields a partition of the global domain $\Omega = \{1, \ldots, n\} = \mathcal{I}_1 \cup \cdots \cup \mathcal{I}_N \cup \Gamma$ where $\Gamma = \Gamma_1 \cup \cdots \cup \Gamma_N$ is the global interface.

Then, eliminating in parallel the interior unknowns following for instance [Section 2, 25] the original system (1.1) reduces to a Schur problem defined on the interface Γ

$$\mathcal{S} u_\Gamma = \widetilde{f}'_\Gamma, \qquad \mathcal{S} = \sum_{i=1}^{N} \mathcal{R}_{\Gamma_i}^T \, \mathcal{S}_i \, \mathcal{R}_{\Gamma_i}, \tag{1.3}$$

where the global Schur matrix $\mathcal{S}$ is SPD and the local Schur matrices $\mathcal{S}_i$ are SPSD. Using the classical index notation for referring to sub-blocks of matrices and vectors, we have $\mathcal{S} = \mathcal{K}_{\Gamma\Gamma} - \sum_{i=1}^{N} \mathcal{K}_{\Gamma\mathcal{I}_i} \mathcal{K}_{\mathcal{I}_i\mathcal{I}_i}^{-1} \mathcal{K}_{\mathcal{I}_i\Gamma}$, $\widetilde{f}'_\Gamma = f_\Gamma - \sum_{i=1}^{N} \mathcal{K}_{\Gamma\mathcal{I}_i} \mathcal{K}_{\mathcal{I}_i\mathcal{I}_i}^{-1} f_{\mathcal{I}_i}$ and $\mathcal{S}_i = \mathcal{K}_{\Gamma_i\Gamma_i} - \mathcal{K}_{\Gamma_i\mathcal{I}_i} \mathcal{K}_{\mathcal{I}_i\mathcal{I}_i}^{-1} \mathcal{K}_{\mathcal{I}_i\Gamma_i}$. From the interface solution u_Γ, the solution in $\mathcal{I}_i$ can be computed as $u_{\mathcal{I}_i} = \mathcal{K}_{\mathcal{I}_i\mathcal{I}_i}^{-1} \left(f_{\mathcal{I}_i} - \mathcal{K}_{\mathcal{I}_i\mathcal{I}_i}^{-1} \mathcal{K}_{\mathcal{I}_i\Gamma} u_\Gamma\right)$.

Algebraically, the problems (1.1) and (1.3) are very similar; their only difference is that even when $\mathcal{K}_i$ is sparse, $\mathcal{S}_i$ is in general a dense matrix (as soon as $\mathcal{K}_i$ is irreducible). Although eliminating the interior unknowns is often associated with specific DDM such as BDD [8, 26] or FETI [13], it is in fact an optional step in the solution of Problem (1.1) and most domain decomposition methods can be applied either directly on $\mathcal{K}$ or, after eliminating the interior unknowns, on $\mathcal{S}$. This elimination step may take time and consume memory, but it allows us to reduce the size and the condition number of the linear system ($\mathcal{S}$) to be solved [5, 27], making it a useful optional preprocessing. Since the theory presented in sections 2 and 3 can be applied to solve either the original problem in (1.1) or the reduced Schur problem in (1.3), we write them in a general form as

$$\mathcal{A} x = b, \qquad \mathcal{A} = \sum_{i=1}^{N} \mathcal{R}_i^T \, \mathcal{A}_i \, \mathcal{R}_i, \tag{1.4}$$

where the global SPD matrix $\mathcal{A}$, the local SPSD matrices $\mathcal{A}_i$, and the restriction matrices $\mathcal{R}_i$ can represent $\mathcal{K}$, $\mathcal{K}_i$ and $\mathcal{R}_{\Omega_i}$ or $\mathcal{S}$, $\mathcal{S}_i$ and $\mathcal{R}_{\Gamma_i}$ when solving (1.1) or (1.3), respectively. When needed, a specific method M will be noted $M/\mathcal{K}$ or $M/\mathcal{S}$ to specify on which problem this method is

applied. In both cases, $\mathcal{A}$ is SPD, assuming that the $\mathcal{A}_i$ are assigned to different computing units, Problem (1.4) can be solved in parallel using the preconditioned conjugate gradient method (PCG).

A good preconditioner $\mathcal{M}$ for (1.4) should have the two following properties: (1) $\mathcal{M}$ is SPD and *close* to $\mathcal{A}^{-1}$, in the sense that the condition number $\kappa(\mathcal{M}\mathcal{A})$ should be as small as possible; (2) it is easy to compute $\mathcal{M}u$ for any vector u (at least much easier than $\mathcal{A}^{-1}u$). DDM are often used to build such preconditioners of the form

$$\mathcal{M}_{aS} = \sum_{i=1}^{N} \mathcal{R}_i^T \widehat{\mathcal{A}}_i^{\dagger} \mathcal{R}_i \tag{1.5}$$

where $\widehat{\mathcal{A}}_i$ is a local problem associated with $\mathcal{A}$ on subdomain i, and † represents a pseudo-inverse.

These preconditioners have been studied for a long time using the abstract Schwarz (aS) theory (see, e.g., [10, 35] for recent overviews). Two particular cases of preconditioners that fit this description are the Neumann-Neumann (NN) preconditioner [26], with $\widehat{\mathcal{A}}_i = D_i^{-1}\mathcal{A}_i D_i^{-1}$, and the Additive Schwarz (AS) preconditioner, with $\widehat{\mathcal{A}}_i = \mathcal{R}_i \mathcal{A} \mathcal{R}_i^T$

$$\mathcal{M}_{NN} = \sum_{i=1}^{N} \mathcal{R}_i^T D_i \mathcal{A}_i^{\dagger} D_i \mathcal{R}_i, \qquad \mathcal{M}_{AS} = \sum_{i=1}^{N} \mathcal{R}_i^T \left(\mathcal{R}_i \mathcal{A} \mathcal{R}_i^T\right)^{-1} \mathcal{R}_i, \tag{1.6}$$

where $(D_i)_{i=1}^N$ is a partition of unity such that $\sum_{i=1}^N \mathcal{R}_i^T D_i \mathcal{R}_i = I_n$ and I_n is the $n \times n$ identity matrix. These two preconditioners are of particular importance, but any other SPSD matrix can be used as the local preconditioner $\widehat{\mathcal{A}}_i$ in (1.5).

Unless $\widehat{\mathcal{A}}_i$ perfectly mimics the global action of $\mathcal{A}$ in subdomain Ω_i, $\kappa(\mathcal{M}_{aS}\mathcal{A})$ may significantly increase with the number N of subdomains, leading to a non scalable numerical method.

Furthermore, if $\widehat{\mathcal{A}}_i$ is singular, the pseudo-inverse is only defined up to an element in its null-space $\ker(\widehat{\mathcal{A}}_i)$. To solve these two problems, a coarse space V_0 such that $\mathcal{R}_i^T \ker(\widehat{\mathcal{A}}_i) \subset V_0$ can be introduced, leading to the deflated aS preconditioner

$$\mathcal{M}_{aS,D} = V_0(V_0^T\mathcal{A}V_0)^{\dagger}V_0^T + (I_n - \mathcal{P}_0)\left(\sum_{i=1}^{N} \mathcal{R}_i^T \widehat{\mathcal{A}}_i^{\dagger} \mathcal{R}_i\right)(I_n - \mathcal{P}_0)^T \tag{1.7}$$

where $\mathcal{P}_0 = V_0(V_0^T\mathcal{A}V_0)^{\dagger}V_0^T\mathcal{A}$ is the $\mathcal{A}$-orthogonal projection onto V_0. A simpler additive two-level preconditioner can also be obtained by just adding the coarse component to the one-level preconditioner

$$\mathcal{M}_{aS,2} = V_0(V_0^T\mathcal{A}V_0)^{\dagger}V_0^T + \sum_{i=1}^{N} \mathcal{R}_i^T \widehat{\mathcal{A}}_i^{\dagger} \mathcal{R}_i. \tag{1.8}$$

While previous works had proposed bounds on the condition number $\kappa(\mathcal{M}\mathcal{A})$ on particular numerical cases, often relying on analytical assumptions, Le Tallec and Vidrascu [25] derived an algebraic bound for a new class of preconditioners, relying on the generalized Rayleigh quotient of two local matrices. These preconditioners are called *generalized NN* in the original article; however, because the generalization consists of handling an approximate matrix, we will instead refer to them as *approximate NN* preconditioners in the present article. The approximation is not related to the use of inexact solvers to compute the preconditioner, but to the use of an approximation matrix

$\widetilde{\mathcal{A}}$ instead of $\mathcal{A}$ in the construction of the preconditioner. The approximate NN preconditioner is in fact an exact algebraic NN preconditioner for $\widetilde{\mathcal{A}}$. Then, this approximate preconditioner is used to accelerate the convergence of PCG applied on the exact matrix $\mathcal{A}$, guaranteeing a convergence towards the actual solution of Equation (1.4).

This class of approximate NN preconditioners generalizes classical NN but does not cover the whole aS class of preconditioners. Note, for instance, that AS cannot be expressed as a NN preconditioner. The first contribution (Section 2) of this article is to extend the result from [25] by using a generic local preconditioner and cover a broader range of aS methods, which we name *approximate deflated aS* methods and consist of all deflated aS methods whose coarse grid consists of the assembly of local components that contain the kernel of some local operators (that are formally introduced below, in Definition 1). Interestingly, the bound we exhibit (Theorem 2) highlights the key position of NN and AS among other local preconditioners in the Schwarz framework: they provide two bounds on the spectrum of the preconditioned operator, and the convergence of any aS local preconditioner can be evaluated by comparing it to these two well-known methods.

This bound depends on generalized Rayleigh quotients which are traditionally estimated using functional analysis. Alternatively, we propose to control these Rayleigh quotients algebraically by building the coarse space using eigenvectors of well chosen generalized eigenproblems (Theorem 10). For that, we follow the Generalized Eigenvalue in the Overlap (GenEO) procedure [33]. This second contribution (Section 3) results in an explicit procedure for building a robust coarse space of any approximate deflated aS method leading to a bound on the condition number (hence on the number of iterations of PCG) independent of the number of subdomains. This result can be readily applied to retrieve or improve the bounds previously obtained via generalized eigenproblems in the particular cases of AS/$\mathcal{K}$ [33], NN/$\mathcal{S}$ [34] and optimized Robin (SORAS/$\mathcal{K}$) [18]. It also generalizes these results to the approximate case. The idea of building a coarse space by solving local eigenproblems in each subdomain was introduced in [15, 29]; it was successfully applied for other DDM such as FETI-DP [12] or BDDC [9] in [22, 23, 24].

The third contribution (Section 4) of this paper is that the application of the considered coarse grid correction in an additive fashion is robust in the approximate AS case (although it is not robust for aS methods in general). The bound we obtain (Theorem 12) can be applied for retrieving the bound obtained in [33], when the coarse correction is applied additively to the AS method on the original matrix (AS/$\mathcal{K}$). When working on the Schur matrix (AS/$\mathcal{S}$) [6], the bound is still valid and leads, as commented in [15], to a smaller coarse space compared to AS/$\mathcal{K}$.

Numerical experiments illustrate our discussion in Section 5. A high performance implementation of the coarse grid correction of one particular, consistently robust method (AS/$\mathcal{S}$) has furthermore been implemented in the high-performance MaPHyS[1] hybrid (direct/iterative) sparse linear solver [2, 3] to eventually assess its performance on a modern parallel computer (Section 5.5) and make this scalable method available to the scientific community.

The paper is organized as follows. Section 2 introduces a new class of approximate (deflated) aS preconditioners and provides a bound on their condition number, which depends on generalized Rayleigh quotients. Applying the GenEO procedure on two well chosen generalized eigenproblems, Section 3 proposes a procedure to explicitly compute the coarse space while bounding these Rayleigh quotients leading to a bound on the condition number (hence on the number of iterations of PCG) independent of the number of subdomains. Section 4 shows that a similar result (and procedure) can be obtained when the coarse grid correction is additively applied, in the case of approximate

[1] See https://gitlab.inria.fr/solverstack/maphys/

AS problems. Numerical experiments illustrate our discussion in Section 5 before concluding in Section 6.

2. Approximate abstract Schwarz preconditioners. In this section, we first define a class of approximate aS preconditioners, which combine a local preconditioner $\widehat{\mathcal{A}}_i$, an approximate matrix $\widetilde{\mathcal{A}}$ and a coarse space V_0 in Section 2.1. We then provide a bound on the condition number of this class of methods in Section 2.2, whose proof is provided in Section 2.3.

2.1. Context.

DEFINITION 1 (Approximate abstract Schwarz preconditioner $\widetilde{\mathcal{M}}_{aS,D}$).
In order to build such a preconditioner for Problem (1.4), we need the three following ingredients:

1. *a set of symmetric positive semi-definite (SPSD) local preconditioners $\widehat{\mathcal{A}}_i$,*
2. *an approximation $\widetilde{\mathcal{A}}$ of $\mathcal{A}$ such that*

$$\exists\ (\widetilde{\mathcal{A}}_i)_{i=1}^N, \quad \widetilde{\mathcal{A}} = \sum_{i=1}^N \mathcal{R}_i^T \widetilde{\mathcal{A}}_i \mathcal{R}_i \ \textit{and}\ \widetilde{\mathcal{A}}_i \ \textit{is SPSD}, \tag{2.1}$$

$$\exists\ \omega_-, \omega_+ > 0, \quad \forall v \in V \quad \omega_-\ v^T \mathcal{A} v \leq v^T \widetilde{\mathcal{A}} v \leq \omega_+\ v^T \mathcal{A} v, \tag{2.2}$$

3. *and a coarse space V_0 such that*

$$\exists\ (V_i^0)_{i=1}^N, \quad V_0 = \sum_{i=1}^N \mathcal{R}_i^T V_i^0 \quad \textit{with} \quad \ker(\widehat{\mathcal{A}}_i) + \ker(\widetilde{\mathcal{A}}_i^{(NN)}) \subset V_i^0, \tag{2.3}$$

where $\widetilde{\mathcal{A}}_i^{(NN)} = D_i^{-1} \widetilde{\mathcal{A}}_i D_i^{-1}$.

We can then define a coarse matrix $\widetilde{\mathcal{A}}_0 = V_0^T \widetilde{\mathcal{A}} V_0$, a coarse projection $\widetilde{\mathcal{P}}_0 = V_0 \widetilde{\mathcal{A}}_0^\dagger V_0^T \widetilde{\mathcal{A}}$ and the approximate aS preconditioner *is then defined as*

$$\widetilde{\mathcal{M}}_{aS,D} = V_0 \widetilde{\mathcal{A}}_0^\dagger V_0^T + (I_n - \widetilde{\mathcal{P}}_0) \left(\sum_{i=1}^N \mathcal{R}_i^T \widehat{\mathcal{A}}_i^\dagger \mathcal{R}_i \right) (I_n - \widetilde{\mathcal{P}}_0)^T. \tag{2.4}$$

Note that the matrix $\widetilde{\mathcal{A}}_i^{(NN)}$ introduced in (2.3) is the local matrix in the approximate NN preconditioner $\widetilde{\mathcal{M}}_{NN,D}$ with the algebraic decomposition from (2.1). The matrices D_i can be any partition of unity as in (1.6). $\widetilde{\mathcal{A}}_i^{(NN)}$ is a scaled version of the local matrix $\widetilde{\mathcal{A}}_i$ in the approximation $\widetilde{\mathcal{A}}$ of $\mathcal{A}$.

When no approximation is used, after a suitable initialization, $\widetilde{\mathcal{M}}_{aS,D}$ can be replaced by $(I_n - \widetilde{\mathcal{P}}_0) \left(\sum_{i=1}^N \mathcal{R}_i^T \widehat{\mathcal{A}}_i^\dagger \mathcal{R}_i \right)$ in the PCG iterations, as noted in [26].

2.2. Convergence result for $\widetilde{\mathcal{M}}_{aS,D}$. In each subdomain, we note $N_i = \#\{j \neq i,\ \mathcal{R}_i \widetilde{\mathcal{A}} \mathcal{R}_j^T \neq 0\}$ the number of neighbors through the connectivity graph of $\widetilde{\mathcal{A}}$. We also define two local subspaces $\widehat{V}_i^\perp$ and $\widetilde{V}_i^\perp$ as the orthogonal spaces of V_i^0 for the inner products inferred by $\widehat{\mathcal{A}}_i$ in range$(\widehat{\mathcal{A}}_i)$ and $\widetilde{\mathcal{A}}_i^{(NN)}$ in range$(\widetilde{\mathcal{A}}_i^{(NN)})$ respectively. Then,

$$\text{range}(\mathcal{R}_i) = \widehat{V}_i^\perp \oplus V_i^0 = \widetilde{V}_i^\perp \oplus V_i^0, \tag{2.5}$$

$$\forall u \in V_i^0,\ \forall v \in \widehat{V}_i^\perp,\ \forall w \in \widetilde{V}_i^\perp \qquad u^T \widehat{\mathcal{A}}_i v = u^T \widetilde{\mathcal{A}}_i^{(NN)} w = 0. \tag{2.6}$$

Finally, for any SPSD matrix $\mathcal{B}$ and vector u, we note $|u|_{\mathcal{B}} = \sqrt{u^T \mathcal{B} u}$ the $\mathcal{B}$-seminorm of u; if $\mathcal{B}$ is SPD, we note it $||u||_{\mathcal{B}}$.

THEOREM 2 (Convergence result for approximate aS).
The condition number of the preconditioned matrix $\widetilde{\mathcal{M}}_{aS,D}\mathcal{A}$ is bounded by

$$\kappa(\widetilde{\mathcal{M}}_{aS,D}\mathcal{A}) \leq \frac{\omega_+}{\omega_-}\left(1 + \max_{1\leq i\leq N} \sup_{v\in\widetilde{V}_i^{\perp}} \frac{|v|^2_{\widehat{\mathcal{A}}_i}}{|v|^2_{\widetilde{\mathcal{A}}_i^{(NN)}}}\right) \max\left(1, \max_{1\leq i\leq N} (N_i+1) \sup_{v\in\widehat{V}_i^{\perp}} \frac{|v|^2_{\widetilde{\mathcal{A}}_i^{(AS)}}}{|v|^2_{\widehat{\mathcal{A}}_i}}\right),$$

where $\widetilde{\mathcal{A}}_i^{(NN)} = D_i^{-1}\widetilde{\mathcal{A}}_i D_i^{-1}$ and $\widetilde{\mathcal{A}}_i^{(AS)} = \mathcal{R}_i\widetilde{\mathcal{A}}\mathcal{R}_i^T$.

We see three factors in this bound:

- The first one, with ω_+ and ω_-, controls the quality of the approximation $\widetilde{\mathcal{A}}$. If no approximation is used, then $\widetilde{\mathcal{A}} = \mathcal{A}$ and $\omega_- = \omega_+ = 1$.
- The second one is a generalized Rayleigh quotient between the local preconditioner $\widehat{\mathcal{A}}_i$ and the approximate NN preconditioner $\widetilde{\mathcal{A}}_i^{(NN)} = D_i^{-1}\widetilde{\mathcal{A}}_i D_i^{-1}$ defined in [25].
- The last one is a generalized Rayleigh quotient between the local preconditioner $\widehat{\mathcal{A}}_i$ and an approximate AS preconditioner $\widetilde{\mathcal{A}}_i^{(AS)} = \mathcal{R}_i\widetilde{\mathcal{A}}\mathcal{R}_i^T$.

As for $\widetilde{\mathcal{A}}_i^{(NN)}$ above with NN, $\widetilde{\mathcal{A}}_i^{(AS)} = \mathcal{R}_i\widetilde{\mathcal{A}}\mathcal{R}_i^T$ is an algebraic generalization of the local matrix in the AS preconditioner in Equation (1.6), built upon the approximation $\widetilde{\mathcal{A}}$ instead of $\mathcal{A}$.

Proof. The proof of Theorem 2 is a direct consequence of lemmas 6 and 8 in Section 2.3, using the definition of

$$\kappa(\widetilde{\mathcal{M}}_{aS,D}\mathcal{A}) = \frac{\lambda_{\max}(\widetilde{\mathcal{M}}_{aS,D}\mathcal{A})}{\lambda_{\min}(\widetilde{\mathcal{M}}_{aS,D}\mathcal{A})}. \qquad \square$$

COROLLARY 3 (Convergence results for approximate AS and approximate NN).
We define the approximate AS and NN preconditioners $\widetilde{\mathcal{M}}_{AS,D}$ and $\widetilde{\mathcal{M}}_{NN,D}$ by replacing $\widehat{\mathcal{A}}_i$ with $\widetilde{\mathcal{A}}_i^{(AS)}$ or $\widetilde{\mathcal{A}}_i^{(NN)}$ respectively in Equation (2.4). We also define $N_c = \max_{1\leq i\leq N}(N_i+1)$.
Then, the condition numbers of $\widetilde{\mathcal{M}}_{NN,D}\mathcal{A}$ and $\widetilde{\mathcal{M}}_{AS,D}\mathcal{A}$ are bounded by

$$\kappa(\widetilde{\mathcal{M}}_{AS,D}\mathcal{A}) \leq \frac{\omega_+}{\omega_-}\left(1 + \max_{1\leq i\leq N} \sup_{v\in\widetilde{V}_i^{\perp}} \frac{|v|^2_{\widetilde{\mathcal{A}}_i^{(AS)}}}{|v|^2_{\widetilde{\mathcal{A}}_i^{(NN)}}}\right) N_c,$$

$$\kappa(\widetilde{\mathcal{M}}_{NN,D}\mathcal{A}) \leq \frac{\omega_+}{\omega_-}\max\left(1, \sup_{v\in\widehat{V}_i^{\perp}} \frac{|v|^2_{\widetilde{\mathcal{A}}_i^{(AS)}}}{|v|^2_{\widetilde{\mathcal{A}}_i^{(NN)}}}\right) N_c.$$

Proof. The proof of Corollary 3 is a consequence of lemmas 6 and 7 for AS, and lemmas 5 and 8 for NN. $\square$

Note that the bound for $\widetilde{\mathcal{M}}_{NN,D}$ in Corollary 3 is the same as in [Theorem 1, 25]. This bound is tighter than the bound obtained by setting $\widehat{\mathcal{A}}_i = \widetilde{\mathcal{A}}_i^{(NN)}$ in Theorem 2; this comes from the fact that the bound in Lemma 5 is also tighter than its generalization in Lemma 6.

The similarity of the bounds for AS and NN in Corollary 3 shows that the convergence of these two methods are governed by the same quantity $\sup_{v\in\widetilde{V}_i^{\perp}} |v|^2_{\widetilde{\mathcal{A}}_i^{(AS)}}/|v|^2_{\widetilde{\mathcal{A}}_i^{(NN)}}$. As a result, with

the same coarse space, we expect the AS/$\mathcal{S}$ method [6] to show the same convergence behavior as the BDD method (NN/$\mathcal{S}$) [26] or its dual counterpart FETI [13]. Although AS require more communication than NN (each subdomain i has to send the matrix block $\mathcal{R}_j\mathcal{R}_i^T\widetilde{\mathcal{A}}_i\mathcal{R}_i\mathcal{R}_j^T$ to each neighbor j) to setup the preconditioner, one advantage of using AS over NN is that the local preconditioner $\widetilde{\mathcal{A}}_i^{(NN)}$ is often singular in some domains while $\widetilde{\mathcal{A}}_i^{(AS)}$ remains SPD, and $\widetilde{\mathcal{A}}_i^{(AS)-1}u_i$ is easier and faster to compute than $\widetilde{\mathcal{A}}_i^{(NN)\dagger}u_i$.

2.3. Proof of Theorem 2. To estimate the condition number of $\widetilde{\mathcal{M}}_{aS,D}\mathcal{A}$, we need to bound the spectrum of this operator from above and below. The lower bound is a consequence of the Stable Decomposition Lemma as stated in [35].

LEMMA 4 (Stable decomposition lemma).
If there exists a constant C_0, local matrices $\mathcal{B}_i$ and extension operators $\mathcal{I}_i$, such that $\ker(\mathcal{B}_i) \subset \ker(\mathcal{I}_i)$ *and every $u \in V$ admits a decomposition*

$$u = \sum_{i=0}^{N} \mathcal{I}_i u_i, \quad \{u_i \in V_i, 0 \leq i \leq N\} \qquad \textit{that satisfies} \qquad \sum_{i=0}^{N} |u_i|^2_{\mathcal{B}_i} \leq C_0^2 \ ||u||^2_{\mathcal{A}}.$$

Then

$$\lambda_{\min}(\mathcal{M}\mathcal{A}) \geq C_0^{-2}, \quad \textit{where} \quad \mathcal{M} = \sum_{i=0}^{N} \mathcal{I}_i \mathcal{B}_i^{\dagger} \mathcal{I}_i^T.$$

Proof. see, e.g., Lemma 2.5 in [35]. ∎

Then, although it is not directly used in the proof of Theorem 2, we first expose in Lemma 5 a lower bound for the spectrum of NN ($\widehat{\mathcal{A}}_i = \widetilde{\mathcal{A}}_i^{(NN)}$) as it provides a good insight on the reason behind the Rayleigh quotients in the bound presented in Lemma 6 for the general case.

LEMMA 5 (Lower bound for the approximate Neumann-Neumann preconditioner).
Let $\widetilde{\mathcal{M}}_{NN,D} = V_0\widetilde{\mathcal{A}}_0^{\dagger}V_0^T + (I_n - \widetilde{\mathcal{P}}_0)\left(\sum_{i=1}^{N} \mathcal{R}_i^T \widetilde{\mathcal{A}}_i^{(NN)^{\dagger}} \mathcal{R}_i\right)(I_n - \widetilde{\mathcal{P}}_0)^T.$
Then,

$$\lambda_{\min}(\widetilde{\mathcal{M}}_{NN,D}\mathcal{A}) \geq \frac{1}{\omega_+}.$$

Proof. This is a consequence of Lemma 4 (see Theorem 1 in [25]). ∎

If, instead of $\widetilde{\mathcal{A}}_i^{(NN)}$, another local preconditioner $\widehat{\mathcal{A}}_i$ is used, there is no change on the bound if we restrict the operators to the coarse space V_0 since the application of the local preconditioner is preceded and followed by projections $(I_n - \widetilde{\mathcal{P}}_0)$ and $(I_n - \widetilde{\mathcal{P}}_0)^T$. However, in the orthogonal of the coarse space, the bound has to change and reflect the difference between $\widetilde{\mathcal{A}}_i^{(NN)}$ and $\widehat{\mathcal{A}}_i$. As is proved in Lemma 6, the lower bound on the spectrum of $\widetilde{\mathcal{M}}_{aS,D}\mathcal{A}$ can be deduced from the bound for $\widetilde{\mathcal{M}}_{NN,D}\mathcal{A}$ in Lemma 5 by adding a correction related to the generalized Rayleigh quotient between $\widetilde{\mathcal{A}}_i^{(NN)}$ and $\widehat{\mathcal{A}}_i$ in the orthogonal of the coarse space.

LEMMA 6 (Lower bound for the approximate abstract Schwarz preconditioner).

$$\lambda_{\min}(\widetilde{\mathcal{M}}_{aS,D}\mathcal{A}) \geq \frac{1}{\omega_+}\left(1+\max_{1\leq i\leq N}\sup_{v\in\widetilde{V}_i^\perp}\frac{|v|^2_{\widehat{\mathcal{A}}_i}}{|v|^2_{\widetilde{\mathcal{A}}_i^{(NN)}}}\right)^{-1}.$$

Proof. We want to split u into a sum of local contributions, while being able to uniformly control the $\widehat{\mathcal{A}}_i$-norm of these contributions u_i with the global $\mathcal{A}$-norm of u to apply Lemma 4. For any u and $i \geq 1$, we decompose $D_i\mathcal{R}_i u = u_i^0 + u_i^\perp$ where $u_i^0 \in V_i^0$ and $u_i^\perp \in \widetilde{V}_i^\perp$. We then define $u_0 = (V_0^T\widetilde{\mathcal{A}}V_0)^\dagger V_0^T\mathcal{A}u$ such that $V_0u_0 = \widetilde{\mathcal{P}}_0 u$. We can use the facts that $\sum_{i=1}^N \mathcal{R}_i^T D_i\mathcal{R}_i = I_n$ and $\sum_{i=0}^N \mathcal{R}_i^T u_i^0 \in V_0 \subset \ker(I_n - \widetilde{\mathcal{P}}_0)$ to obtain the decomposition

$$\begin{aligned}
u &= \widetilde{\mathcal{P}}_0 u + (I_n - \widetilde{\mathcal{P}}_0)u = V_0u_0 + (I_n - \widetilde{\mathcal{P}}_0)\sum_{i=1}^N \mathcal{R}_i^T D_i\mathcal{R}_i u \\
&= V_0u_0 + (I_n - \widetilde{\mathcal{P}}_0)\sum_{i=1}^N \mathcal{R}_i^T(u_i^0 + u_i^\perp) = V_0u_0 + (I_n - \widetilde{\mathcal{P}}_0)\sum_{i=1}^N \mathcal{R}_i^T u_i^\perp \\
&= \sum_{i=0}^N \mathcal{I}_i u_i \quad \text{where} \quad \mathcal{I}_0 = V_0, \quad \mathcal{I}_i = (I_n - \widetilde{\mathcal{P}}_0)\mathcal{R}_i \quad \text{and} \quad u_i = u_i^\perp.
\end{aligned}$$

Since $\widetilde{\mathcal{P}}_0$ is a $\widetilde{\mathcal{A}}$-orthogonal projection, it holds that:

$$|u_0|^2_{\widetilde{\mathcal{A}}_0} = |u_0|^2_{V_0^T\widetilde{\mathcal{A}}V_0} = |V_0u_0|^2_{\widetilde{\mathcal{A}}} = |\widetilde{\mathcal{P}}_0 u|^2_{\widetilde{\mathcal{A}}} \leq |u|^2_{\widetilde{\mathcal{A}}} \tag{2.7}$$

Let

$$C = \max_{1\leq i\leq N}\sup_{v\in\widetilde{V}_i^\perp}\frac{|v|^2_{\widehat{\mathcal{A}}_i}}{|v|^2_{\widetilde{\mathcal{A}}_i^{(NN)}}} = \max_{1\leq i\leq N}\sup_{v\in\widetilde{V}_i^\perp}\frac{|v|^2_{\widehat{\mathcal{A}}_i}}{|v|^2_{D_i^{-1}\widetilde{\mathcal{A}}_i D_i^{-1}}}.$$

We can then use equations (2.6), (2.1) and (2.7):

$$\begin{aligned}
&|u_i^\perp|^2_{\widehat{\mathcal{A}}_i} \leq C|u_i^\perp|^2_{D_i^{-1}\widetilde{\mathcal{A}}_i D_i^{-1}} \leq C|u_i^\perp + u_i^0|^2_{D_i^{-1}\widetilde{\mathcal{A}}_i D_i^{-1}} = C|\mathcal{R}_i u|^2_{\widetilde{\mathcal{A}}_i}, \\
&\sum_{i=1}^N |u_i^\perp|^2_{\widehat{\mathcal{A}}_i} \leq C\sum_{i=1}^N |\mathcal{R}_i u|^2_{\widetilde{\mathcal{A}}_i} = C|u|^2_{\sum_{i=1}^N \mathcal{R}_i^T\widetilde{\mathcal{A}}_i\mathcal{R}_i} = C|u|^2_{\widetilde{\mathcal{A}}}, \\
&|u_0|^2_{\widetilde{\mathcal{A}}_0} + \sum_{i=1}^N |u_i^\perp|^2_{\widehat{\mathcal{A}}_i} \leq (1+C)\,|u|^2_{\widetilde{\mathcal{A}}} \leq \omega_+(1+C)|u|^2_{\mathcal{A}},
\end{aligned} \tag{2.8}$$

and the local norms are controlled by the global norm. Then, applying Lemma 4, we get

$$\lambda_{\min}(\widetilde{\mathcal{M}}_{aS,D}\mathcal{A}) \geq \frac{1}{\omega_+}\left(1+\max_{1\leq i\leq N}\sup_{v\in\widetilde{V}_i^\perp}\frac{|v|^2_{\widehat{\mathcal{A}}_i}}{|v|^2_{\widetilde{\mathcal{A}}_i^{(NN)}}}\right)^{-1}. \qquad \square$$

Now that we proved a lower bound for the spectrum of $\widetilde{\mathcal{M}}_{NN,D}\mathcal{A}$, we will prove an upper bound in Lemma 8. We first recall a classic upper bound for AS preconditioners in Lemma 7 since it explains the origin of the Rayleigh quotient in the bound for the general case.

LEMMA 7 (Upper bound for the approximate Additive Schwarz preconditioner).
Let $\widetilde{\mathcal{M}}_{AS,D} = V_0\widetilde{\mathcal{A}}_0^{\dagger}V_0^T + (I_n - \widetilde{\mathcal{P}}_0)\left(\sum_{i=1}^{N}\mathcal{R}_i^T\widetilde{\mathcal{A}}_i^{(AS)^{-1}}\mathcal{R}_i\right)(I_n - \widetilde{\mathcal{P}}_0)^T$.
Then,

$$\lambda_{\max}(\widetilde{\mathcal{M}}_{AS,D}\mathcal{A}) \leq \frac{1}{\omega_-}\max_{1\leq i\leq N}(N_i+1).$$

Proof. This lemma is a particular case of Lemma 8 which is proven below. ∎

LEMMA 8 (Upper bound for the approximate abstract Schwarz preconditioner).

$$\lambda_{\max}(\widetilde{\mathcal{M}}_{aS,D}\mathcal{A}) \leq \frac{1}{\omega_-}\max\left(1, \max_{1\leq i\leq N}(N_i+1)\sup_{v\in\widehat{V}_i^{\perp}}\frac{|v|^2_{\widetilde{\mathcal{A}}_i^{(AS)}}}{|v|^2_{\widehat{\mathcal{A}}_i}}\right).$$

Proof. First, let us remark that

$$\widetilde{\mathcal{M}}_{aS,D}\widetilde{\mathcal{A}}u = V_0\widetilde{\mathcal{A}}_0^{\dagger}V_0^T\widetilde{\mathcal{A}}u + (I_n - \widetilde{\mathcal{P}}_0)\sum_{i=1}^{N}\mathcal{R}_i^T\widehat{\mathcal{A}}_i^{\dagger}\mathcal{R}_i(I_n - \widetilde{\mathcal{P}}_0)^T\widetilde{\mathcal{A}}u = u_0 + (I_n - \widetilde{\mathcal{P}}_0)\sum_{i=1}^{N}\mathcal{R}_i^T u_i$$

where $u_0 = \widetilde{\mathcal{P}}_0 u$ and u_i is the orthogonal projection of $\widehat{\mathcal{A}}_i^{\dagger}\mathcal{R}_i(I_n - \widetilde{\mathcal{P}}_0)^T\widetilde{\mathcal{A}}u$ onto range$(\widehat{\mathcal{A}}_i)$ along $\ker(\widehat{\mathcal{A}}_i) \subset V_i^0 \subset \ker\left[(I_n - \widetilde{\mathcal{P}}_0)\mathcal{R}_i^T\right]$.

As a consequence, $u_i \in \widehat{V}_i^{\perp}$:

$$u_i^T\widehat{\mathcal{A}}_i V_i^0 = u^T\widetilde{\mathcal{A}}(I_n - \widetilde{\mathcal{P}}_0)\mathcal{R}_i^T\widehat{\mathcal{A}}_i^{\dagger}\widehat{\mathcal{A}}_i V_i^0 = u^T\widetilde{\mathcal{A}}(I_n - \widetilde{\mathcal{P}}_0)\mathcal{R}_i^T V_i^0 = 0.$$

Then,

$$\begin{aligned}|\widetilde{\mathcal{M}}_{aS,D}\widetilde{\mathcal{A}}u|^2_{\widetilde{\mathcal{A}}} &= |u_0|^2_{\widetilde{\mathcal{A}}} + |(I_n - \widetilde{\mathcal{P}}_0)\sum_{i=1}^{N}\mathcal{R}_i^T u_i|^2_{\widetilde{\mathcal{A}}} \leq |u_0|^2_{\widetilde{\mathcal{A}}} + |\sum_{i=1}^{N}\mathcal{R}_i^T u_i|^2_{\widetilde{\mathcal{A}}}\\ &\leq |u_0|^2_{\widetilde{\mathcal{A}}} + \sum_{i=1}^{N}(N_i+1)|\mathcal{R}_i^T u_i|^2_{\widetilde{\mathcal{A}}} = |u_0|^2_{\widetilde{\mathcal{A}}} + \sum_{i=1}^{N}(N_i+1)|u_i|^2_{\mathcal{R}_i\widetilde{\mathcal{A}}\mathcal{R}_i^T}\end{aligned}$$

where we used the fact that

$$\begin{aligned}0 \leq \sum_{\substack{1\leq i,j\leq N\\ \mathcal{R}_i^T\widetilde{\mathcal{A}}\mathcal{R}_j\neq 0}}|\mathcal{R}_i^T u_i - \mathcal{R}_j^T u_j|^2_{\widetilde{\mathcal{A}}} &= 2\left(\sum_{\substack{1\leq i,j\leq N\\ \mathcal{R}_i^T\widetilde{\mathcal{A}}\mathcal{R}_j\neq 0}}|\mathcal{R}_i^T u_i|^2_{\widetilde{\mathcal{A}}} - \sum_{\substack{1\leq i,j\leq N\\ \mathcal{R}_i^T\widetilde{\mathcal{A}}\mathcal{R}_j\neq 0}}u_i^T\mathcal{R}_i\widetilde{\mathcal{A}}\mathcal{R}_j^T u_j\right)\\ &\leq 2\left(\sum_{i=1}^{N}(N_i+1)|\mathcal{R}_i^T u_i|^2_{\widetilde{\mathcal{A}}} - |\sum_{i=1}^{N}\mathcal{R}_i^T u_i|^2_{\widetilde{\mathcal{A}}}\right). \qquad (2.9)\end{aligned}$$

Let us define

$$C = \max\left(1, \max_{1\leq i\leq N}(N_i+1)\sup_{v\in\widehat{V}_i^{\perp}}\frac{|v|^2_{\widetilde{\mathcal{A}}_i^{(AS)}}}{|v|^2_{\widehat{\mathcal{A}}_i}}\right) = \max\left(1, \max_{1\leq i\leq N}(N_i+1)\sup_{v\in\widehat{V}_i^{\perp}}\frac{|v|^2_{\mathcal{R}_i\widetilde{\mathcal{A}}\mathcal{R}_i^T}}{|v|^2_{\widehat{\mathcal{A}}_i}}\right).$$

We can now write

$$|\widetilde{\mathcal{M}}_{aS,D}\widetilde{\mathcal{A}}u|^2_{\widetilde{\mathcal{A}}} \leq C|u_0|^2_{\widetilde{\mathcal{A}}} + C\sum_{i=1}^{N}|u_i|^2_{\widehat{\mathcal{A}}_i} = Cu^T\widetilde{\mathcal{P}}_0^T\widetilde{\mathcal{A}}u_0 + C\sum_{i=1}^{N}u^T\widetilde{\mathcal{A}}(I_n-\widetilde{\mathcal{P}}_0)\mathcal{R}_i^T\widehat{\mathcal{A}}_i^\dagger\widehat{\mathcal{A}}_i u_i$$
$$= Cu^T\widetilde{\mathcal{A}}\widetilde{\mathcal{M}}_{aS,D}\widetilde{\mathcal{A}}u \leq C|u|_{\widetilde{\mathcal{A}}}|\widetilde{\mathcal{M}}_{aS,D}\widetilde{\mathcal{A}}u|_{\widetilde{\mathcal{A}}}$$
$$|\widetilde{\mathcal{M}}_{aS,D}\widetilde{\mathcal{A}}u|_{\widetilde{\mathcal{A}}} \leq C|u|_{\widetilde{\mathcal{A}}},$$

and use the same strategy as in [25] to obtain our result:

$$\lambda_{\max}(\widetilde{\mathcal{M}}_{aS,D}\mathcal{A}) = \max_{v\in V}\frac{|v|^2_{\mathcal{A}}}{|v|^2_{\widetilde{\mathcal{M}}^{-1}_{aS,D}}} \leq \max_{v\in V}\frac{1}{\omega_-}\frac{|v|^2_{\widetilde{\mathcal{A}}}}{|v|^2_{\widetilde{\mathcal{M}}^{-1}_{aS,D}}} \leq \max_{v\in V}\frac{1}{\omega_-}\frac{|\widetilde{\mathcal{M}}_{aS,D}\widetilde{\mathcal{A}}v|_{\widetilde{\mathcal{A}}}}{|v|_{\widetilde{\mathcal{A}}}} \leq \frac{C}{\omega_-},$$
$$\lambda_{\max}(\widetilde{\mathcal{M}}_{aS,D}\mathcal{A}) \leq \frac{1}{\omega_-}\max\left(1, \max_{1\leq i\leq N}(N_i+1)\sup_{v\in\widehat{V}_i^\perp}\frac{|v|^2_{\widetilde{\mathcal{A}}_i^{(AS)}}}{|v|^2_{\widehat{\mathcal{A}}_i}}\right). \qquad \square$$

3. Building the coarse space via generalized eigenproblems. The bound in Theorem 2 has originally been estimated through functional analysis after a coarse space has been chosen. A more algebraic approach is to build the coarse space V_0 by solving a generalized eigenproblem in each subdomain in order to control the Rayleigh quotient as proposed by [33, 34] for AS/$\mathcal{K}$ and NN/$\mathcal{S}$, respectively. This approach has also been successfully applied to other aS variants such as the SORAS method [18], in which case two eigenproblems are needed. The case where the correction is applied additively as in [11, 15, 33] for AS is treated in Section 4.

The connection between the GenEO method and Theorem 2 comes from the following lemma:

LEMMA 9 (Bound on the Rayleigh quotient).
Let $\mathcal{B}$ be a SPSD matrix, $\mathcal{C}$ a SPD matrix and $\eta > 0$ be a parameter.
If $V_\eta = span\big(\{p, \quad \mathcal{B}p = \lambda\mathcal{C}p, \quad \lambda \leq \eta\}\big)$ and $V_\eta^{\perp_\mathcal{B}} = \{u \in range(\mathcal{B}), \forall v \in V_\eta, u^T\mathcal{B}v = 0\}$,

$$then \quad \sup_{u\in V_\eta^{\perp_\mathcal{B}}}\frac{|u|^2_{\mathcal{C}}}{|u|^2_{\mathcal{B}}} \leq \frac{1}{\eta}.$$

Proof. Since $\mathcal{C}$ is SPD, the generalized eigenproblem $\mathcal{B}p = \lambda\mathcal{C}p$ has solutions (λ_k, p_k) with $p_k^T\mathcal{C}p_l = \delta_{kl}$ and $p_k^T\mathcal{B}p_l = \lambda_k\delta_{kl}$.

Now, let $u \in V_\eta^{\perp_\mathcal{B}}$. We can project u on the basis $(p_k)_k$: $u = \sum_k \alpha_k p_k$.

If k is such that $\lambda_k \leq \eta$, then $p_k \in V_\eta$ and $0 = u^T\mathcal{B}p_k = \lambda_k\alpha_k$. As a consequence, $\alpha_k = 0$ because if $\lambda_k = 0$, $p_k \in \ker(\mathcal{B}) = \big(\text{range}(\mathcal{B})\big)^\perp \perp u$ and $\alpha_k = u^T p_k = 0$. This leads to

$$\frac{|u|^2_{\mathcal{C}}}{|u|^2_{\mathcal{B}}} = \frac{\sum_{\lambda_k>\eta}\alpha_k^2}{\sum_{\lambda_k>\eta}\lambda_k\alpha_k^2} \leq \frac{1}{\eta}. \qquad \square$$

Following the GenEO methodology, we propose to build the coarse space V_0 by solving two generalized eigenproblems to control the condition number of approximate aS preconditioners through two parameters $\alpha > 0$ and $\beta \geq 1$.

THEOREM 10 (Condition number of aS preconditioners). *If $\widehat{\mathcal{A}}_i$ is SPD and the coarse space is defined as $V_0 = \sum_{i=1}^{N} \mathcal{R}_i^T V_i^0$ with*

$$V_i^0 = span\big(\{p_k^i, \quad \widetilde{\mathcal{A}}_i^{(NN)} p_k^i \;=\; \lambda_k^i \widehat{\mathcal{A}}_i p_k^i, \quad \lambda_k^i \le \alpha^{-1}\}$$
$$\cup\, \{p_k^i, \quad \widehat{\mathcal{A}}_i p_k^i \;=\; \lambda_k^i \widetilde{\mathcal{A}}_i^{(AS)} p_k^i, \quad \lambda_k^i \le (N_i+1)\beta^{-1}\}\big)$$

then, we can bound the condition number

$$\kappa(\widetilde{\mathcal{M}}_{aS,D}\mathcal{A}) \le \frac{\omega_+}{\omega_-}\,(1+\alpha)\,\beta.$$

Proof. Using Lemma 9 and the definition of $\widetilde{V}_i^{\perp}$ and $\widehat{V}_i^{\perp}$ in 2.2, we can bound the Rayleigh quotients

$$\sup_{v\in\widetilde{V}_i^{\perp}} \frac{|v|^2_{\widehat{\mathcal{A}}_i}}{|v|^2_{\widetilde{\mathcal{A}}_i^{(NN)}}} \le \alpha, \qquad\qquad \sup_{v\in\widehat{V}_i^{\perp}} \frac{|v|^2_{\widetilde{\mathcal{A}}_i^{(AS)}}}{|v|^2_{\widehat{\mathcal{A}}_i}} \le \frac{\beta}{N_i+1}.$$

Replacing these bounds in Theorem 2 gives the result. ∎

COROLLARY 11. *In the NN or AS cases, for any $\alpha \ge 1$, we can define*

$$V_i^0 = span\big(\{p_k^i, \quad \widetilde{\mathcal{A}}_i^{(NN)} p_k^i \;=\; \lambda_k^i \widetilde{\mathcal{A}}_i^{(AS)} p_k^i, \quad \lambda_k^i \le \alpha^{-1}\}\big).$$

Then, Corollary 3 and Lemma 9 give

$$\kappa(\widetilde{\mathcal{M}}_{AS,D}\mathcal{A}) \le \frac{\omega_+}{\omega_-}\,(1+\alpha)\;\; N_c, \qquad\qquad \kappa(\widetilde{\mathcal{M}}_{NN,D}\mathcal{A}) \le \frac{\omega_+}{\omega_-}\;\alpha\; N_c.$$

If $\alpha^{-1} = \min_{\lambda_k^i \neq 0}(\lambda_k^i)$, then $V_i^0 = \ker(\widetilde{\mathcal{A}}_i^{(NN)}) = D_i \ker(\widetilde{\mathcal{A}}_i)$ and the resulting coarse space for NN is exactly the same as in the BDD algorithm.

With small variations in the generalized eigenproblems considered, Theorem 10 and Corollary 11 retrieve or improve previous GenEO results and generalize them to the approximate case: AS/$\mathcal{K}$ [32, 33], NN/$\mathcal{S}$ [34] and SORAS [18].

4. Additive Coarse Correction.

4.1. Context. The preconditioner $\widetilde{\mathcal{M}}_{aS,D}$ separates the part of the solution that is in V_0 (on which a direct coarse solve is performed through $\widetilde{\mathcal{A}}_0^{\dagger}$), from its $\widetilde{\mathcal{A}}$-orthogonal part (on which the local preconditioner $\mathcal{M}_{aS} = \sum_{i=1}^{N} \mathcal{R}_i^T \widehat{\mathcal{A}}_i^{\dagger} \mathcal{R}_i$ is used to accelerate convergence). Eigenvalues or Rayleigh quotients λ corresponding to vectors in the coarse space V_0 are shifted to 1 by the coarse solve, and to 0 by the projection steps $(I_n - \widetilde{\mathcal{P}}_0)$ and $(I_n - \widetilde{\mathcal{P}}_0)^T$, so the overall effect of the deflated preconditioner is to shift them to 1 exactly. If we skip these projection steps, we obtain an approximate additive two-level preconditioner $\widetilde{\mathcal{M}}_{aS,2}$ similar to $\mathcal{M}_{aS,2}$ presented in Equation (1.8). In this case, without the projection steps eigenvalues are shifted to $1+\lambda$. As a result, this coarse correction applied on big eigenvalues only makes them bigger, thus hampering convergence. This additive coarse correction can only be effective to tackle the lower part of the spectrum since small eigenvalues $\lambda \ll 1$ are shifted to $1+\lambda \approx 1$.

The one-level AS method has already an upper bound on the spectrum (see Lemma 7), and only the lower bound needs to be recovered, making it an ideal candidate for an additive coarse correction. In this section, we show that in the approximate AS case, when $\widehat{\mathcal{A}}_i = \widetilde{\mathcal{A}}_i^{(AS)} = \mathcal{R}_i \widetilde{\mathcal{A}} \mathcal{R}_i^T$, the projection steps can be removed without losing robustness. Namely, we still have a bound for the condition number of the additive two-level AS method independent of the number of subdomains.

THEOREM 12 (Condition number of the 2-level approximate AS preconditioner).

Let $\mathcal{M}_{AS,2} = V_0 \widetilde{\mathcal{A}}_0^{\dagger} V_0^T + \sum_{i=1}^{N} \mathcal{R}_i^T \widetilde{\mathcal{A}}_i^{(AS)^{-1}} \mathcal{R}_i$ *and* $N_c = \max_{1 \le i \le N} (N_i + 1)$.

Then, we can bound the condition number

$$\kappa(\mathcal{M}_{AS,2}\mathcal{A}) \le \frac{\omega_+}{\omega_-} \left[N_c + 1 + (N_c + 2) \max_{1 \le i \le N} \sup_{v \in \widetilde{V}_i^{\perp}} \frac{|v|^2_{\widetilde{\mathcal{A}}_i^{(AS)}}}{|v|^2_{\widetilde{\mathcal{A}}_i^{(NN)}}} \right] (N_c + 1).$$

For any $\alpha > 0$, *if we choose*

$$V_i^0 = span\big(\{p_k^i, \quad \widetilde{\mathcal{A}}_i^{(NN)} p_k^i = \lambda_k^i \widetilde{\mathcal{A}}_i^{(AS)} p_k^i, \quad \lambda_k^i \le \alpha^{-1}\}\big),$$

it holds that

$$\kappa(\mathcal{M}_{AS,2}\mathcal{A}) \le \frac{\omega_+}{\omega_-} \left[N_c + 1 + \alpha(N_c + 2) \right] (N_c + 1).$$

Theorem 12 generalizes [Theorem 4.40, 33] to the approximate case, while improving the bound.

A spectral coarse space composed of eigenvectors of a generalized eigenproblem was earlier proposed in [11, 15]. In those studies, the authors also discuss the analytical and numerical interest of using AS,2/$\mathcal{S}$ instead of the more traditional AS,2/$\mathcal{K}$ to reduce the size of the coarse space. In comparison, our method is more algebraic in the sense that it does not need a stable interpolation operator, nor the mass matrix.

Proof. If we apply Lemma 7 without a coarse space and consider V_0 as another subdomain in the decomposition, we get

$$\lambda_{\max}(\mathcal{M}_{AS,2}\mathcal{A}) \le \frac{1}{\omega_-}(N_c + 1).$$

The lower bound is a consequence of Lemma 4. We define $u_i^0 \in V_i^0$ and $u_i^{\perp} \in \widetilde{V}_i^{\perp}$ such that $D_i \mathcal{R}_i u = u_i^0 + u_i^{\perp}$ as in the proof of Lemma 6. We now introduce u_0 such that $V_0 u_0 = \sum_{i=1}^{N} \mathcal{R}_i^T u_i^0$, and $u = V_0 u_0 + \sum_{i=1}^{N} \mathcal{R}_i^T u_i^{\perp}$.

We get from Equation (2.8) that

$$\sum_{i=1}^{N} |u_i^{\perp}|^2_{\widetilde{\mathcal{A}}_i^{(AS)}} = \sum_{i=1}^{N} |u_i^{\perp}|^2_{\widehat{\mathcal{A}}_i} \le C |u|^2_{\widetilde{\mathcal{A}}} \qquad \text{with} \qquad C = \max_{1 \le i \le N} \sup_{v \in \widetilde{V}_i^{\perp}} \frac{|v|^2_{\widetilde{\mathcal{A}}_i^{(AS)}}}{|v|^2_{\widetilde{\mathcal{A}}_i^{(NN)}}} = \frac{|v|^2_{\mathcal{R}_i \widetilde{\mathcal{A}} \mathcal{R}_i^T}}{|v|^2_{D_i^{-1} \widetilde{\mathcal{A}}_i D_i^{-1}}}.$$

Then, we can use the same method as in Equation (2.9):

$$|u_0|^2_{\widetilde{\mathcal{A}}} = |u - \sum_{i=1}^{N} \mathcal{R}_i^T u_i^\perp|^2_{\widetilde{\mathcal{A}}} \leq (N_c+1)\left(|u|^2_{\widetilde{\mathcal{A}}} + \sum_{i=1}^{N} |\mathcal{R}_i^T u_i^\perp|^2_{\widetilde{\mathcal{A}}}\right)$$

$$= (N_c+1)\left(|u|^2_{\widetilde{\mathcal{A}}} + \sum_{i=1}^{N} |u_i^\perp|^2_{\mathcal{R}_i\widetilde{\mathcal{A}}\mathcal{R}_i^T}\right) \leq (N_c+1)(1+C)|u|^2_{\widetilde{\mathcal{A}}}$$

$$|u_0|^2_{\widetilde{\mathcal{A}}} + \sum_{i=1}^{N} |u_i^\perp|^2_{\mathcal{R}_i\widetilde{\mathcal{A}}\mathcal{R}_i^T} \leq [N_c+1+(N_c+2)C]\,|u|^2_{\widetilde{\mathcal{A}}} \leq \omega_+\,[N_c+1+(N_c+2)C]\,|u|^2_{\mathcal{A}}.$$

We then use Lemma 4 with $\mathcal{I}_0 = V_0$, $\mathcal{I}_i = \mathcal{R}_i^T$ and $\mathcal{B}_i = \mathcal{I}_i^T \widetilde{\mathcal{A}} \mathcal{I}_i$ to get the bound

$$\lambda_{\min}(\mathcal{M}_{AS,2}\mathcal{A}) \geq \frac{1}{\omega_+}\left[N_c+1+(N_c+2)\max_{1\leq i\leq N}\sup_{v\in\widetilde{V}_i^\perp}\frac{|v|^2_{\mathcal{R}_i\widetilde{\mathcal{A}}\mathcal{R}_i^T}}{|v|^2_{\widetilde{\mathcal{A}}_i^{(NN)}}}\right]^{-1}.$$

We can then conclude with Lemma 9. ∎

5. Numerical experiments.

5.1. Experimental setup. The methods introduced in sections 2, 3 and 4 are tested on a problem similar to what is presented in [33]. We use the Finite Element Method (FEM) with Q1 elements to solve a heterogeneous diffusion equation $\nabla \cdot (k\nabla u) = 1$ in a 3D stratified medium. The domain $[0, N] \times [0, 6] \times [0, 1]$ is discretized on a regular mesh of $(5N + 1) \times 31 \times 6$ nodes. The domain is divided into N identical subdomains along the first axis. Along the second axis, it is divided into 10 layers (of $5N \times 3 \times 5$ elements each) of alternating conductivity $k = 1$ and $k = K$ (K is a heterogeneity parameter). A Dirichlet boundary condition is applied on the left of the domain ($x = 0$), a Neumann condition on every other boundary. Using a FEM discretization on each subdomain gives rise naturally to a set of local SPSD matrices and a global matrix that is SPD. The geometry and 1D partitioning of this test case are chosen so as to emphasize the effects of using a coarse grid correction: indeed, without a coarse correction, the number of iterations grows as $O(N^{1/d})$ where d is the dimension of the partitioning. Using a 3D partitioning of the global domain, one would need more than 7M subdomains (192^3) to illustrate the same effect as in the experiments presented here with a 1D partitioning and 192 subdomains. The layered structure of the domain is introduced to deteriorate the condition number of the local subproblems. Since all subdomains (except the first and last ones) are identical, the bound on the condition number of the method in Theorem 2 is independent of N if at least the kernels of $\widetilde{\mathcal{A}}_i^{(NN)}$ and $\widehat{\mathcal{A}}_i$ are included in V_i^0; a coarse space that only includes these kernels (as in BDD for instance) thus yields a method that can be considered as robust in this regard, while being considerably simpler to compute than the coarse space proposed in this article. However, the condition number still depends on the inverse of the smallest eigenvalues not included in the coarse space, which can be quite close to 0 if the local problems are ill-conditioned (*i.e.*, if K is big). As a result, the condition number, although independent of N, can still be too large for the iterative solver to converge in a reasonable number of iterations. Building the coarse space by solving the generalized eigenproblems as proposed in Section 3 yields a more robust method in the sense that the condition number of the method can be controlled independently of both N, K, and the particular choice of a local preconditioner. We consider three aS methods: the AS and NN preconditioners introduced in Equation (1.6) and a

Shifted (Sh) preconditioner whose local matrix is obtained by shifting the diagonal of $\widetilde{\mathcal{A}}_i$ by 1 to remove its potential singularity: $\widetilde{\mathcal{M}}_{Sh} = \sum_{i=1}^{N} \mathcal{R}_i^T (\widetilde{\mathcal{A}}_i + I_{n_i})^\dagger \mathcal{R}_i$ where I_{n_i} is the identity matrix of same size as $\mathcal{A}_i$. If built on the Schur matrix, $\widetilde{\mathcal{M}}_{Sh}$ is a (non-optimized) Robin preconditioner. The optimization of the Robin condition as proposed in [16] is not considered here as it is out of the scope of this paper. It is introduced as an example of a more generic aS preconditioner than AS and NN; as such, two generalized eigenproblems need to be solved to compute the coarse space for Sh as opposed to only one for AS and NN. Each of these method is assessed with $\mathcal{A} = \mathcal{K}$ or $\mathcal{A} = \mathcal{S}$. Equation (1.4) can therefore either result from:

- the FEM discretization (1.1) of the global problem, in which case the preconditioner is said to be applied on the original matrix $\mathcal{K}$ and the abstract Schwarz method is noted aS/$\mathcal{K}$;
- or the substructuring system (1.3) obtained by eliminating the interior variables from Equation (1.1), in which case the preconditioner is said to be applied on the Schur matrix $\mathcal{S}$ and the method is noted aS/$\mathcal{S}$.

We study the numerical behaviour of these methods under the constraint of a bounded condition number or an imposed coarse space size in sections 5.2 and 5.3, respectively. We then study the approximate case with an empirical approach in Section 5.4, using a so-called *sparsification* technique. Our numerical results overall confirm [11, 15] regarding the numerical interest of using AS,2/$\mathcal{S}$ instead of the more traditional AS,2/$\mathcal{K}$ method to reduce the size of the coarse space. Section 5.5 eventually illustrates the parallel behavior of that promising variant.

The partition of unity D_i is computed using the diagonal values of $\mathcal{A}_i$. The condition numbers of the preconditioned matrices are estimated using the eigenvalues of the tridiagonal Lanczos matrix computed during the PCG iterations (see, e.g., [14]). The stopping criterion is based on the normwise backward error $||b - \mathcal{A}x_k||/||b|| \leq 10^{-6}$.

5.2. Imposing an *a priori* bound on the condition number. We proved in Section 3 that it is possible to control the condition number $\kappa(\widetilde{\mathcal{M}}_{aS,D}\mathcal{A})$ of aS methods through some parameters α and β. For now, we do not use any approximation (whose effects are the object of Section 5.4), hence $\widetilde{\mathcal{A}}_i = \mathcal{A}_i$ and $\omega_- = \omega_+ = 1$. In order to compare the three methods, we first choose a bound χ and then we choose α and β such that $\kappa \leq \chi$:

- for AS (resp. NN), Corollary 11 states that $\kappa \leq (1+\alpha)N_c$ (resp. $\kappa \leq \alpha N_c$). We choose $\alpha = \chi/N_c - 1$ (resp. $\alpha = \chi/N_c$).
- for Sh (or any other aS preconditioner), Theorem 10 states that $\kappa \leq (1+\alpha)\beta$ and we choose $\alpha = \sqrt{1/4+\chi} - 1/2$ and $\beta = \sqrt{1/4+\chi} + 1/2$.

When we do not impose an upper bound ($\chi = \infty$), no coarse space is used and results are presented only for AS and Sh. We observe (Figure 1) that the condition number κ grows quadratically with the number of subdomains N and that the number of iterations to reach convergence (Figure 2) is proportional to the number of subdomains (note the log scale for the x-axis). This lack of scalability is the main motivation for using a two-level method. We also note that, without a coarse space, our AS preconditioner outperforms the Sh preconditioner, especially when the heterogeneity K is high: the AS preconditioner performs a more appropriate local solve than the very basic Sh preconditioner. As expected, the condition number is also lower when working on the Schur matrix $\mathcal{S}$ instead of $\mathcal{K}$, since all the interior unknowns are solved using a direct method and do not appear anymore in the iterative process.

When we impose an upper bound on the condition number ($\chi = 10,000$ or $\chi = 100$), we observe that the condition number κ does indeed drop below the prescribed bound χ, independently of the number of subdomains N, the local preconditioner AS, NN or Sh, the heterogeneity K and the

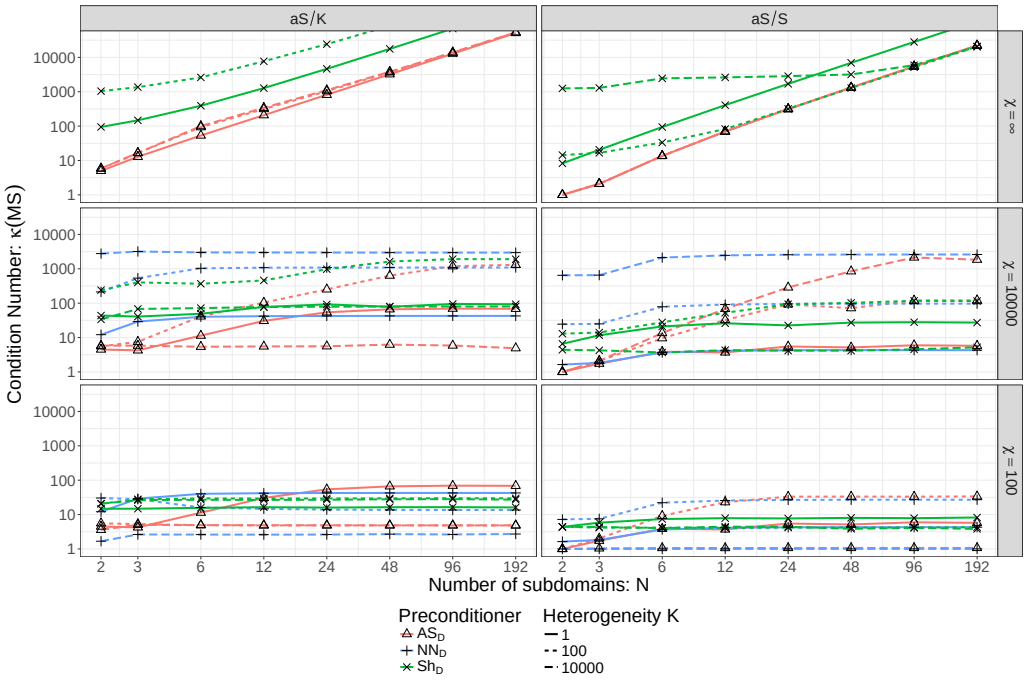

FIG. 1. *Imposing an* a priori *bound χ on the condition number using deflation. Whatever the chosen target χ, we ensure that the condition number of the iterative problem $\kappa(\mathcal{MA})$ remains below χ. Each preconditioner (AS, NN, Sh) can be applied either on the original matrix $\mathcal{K}$ (aS/$\mathcal{K}$), left, or in a substructuring context on $\mathcal{S}$ (aS/$\mathcal{S}$), right.*

choice of operating on $\mathcal{K}$ or $\mathcal{S}$. However, this *a priori* control on the condition number comes at the expense of having to use a direct solve on a coarse space V_0 whose dimension can be quite large. Each subdomain computes a local coarse space V_i^0 of dimension $n_v^{(i)}$ (Figure 3) and the size of the global coarse space therefore grows linearly with the number of subdomains. Since without deflation ($\chi = \infty$) the Sh preconditioner applied to the original matrix $\mathcal{K}$ does not perform very well in the heterogeneous case, the size of the coarse space necessary to obtain a condition number below the target χ is very large (up to 87 vectors per subdomain). However, using a better local preconditioner such as AS or NN can greatly reduce the size of the coarse space, as well as working on the Schur matrix $\mathcal{S}$ instead of $\mathcal{K}$.

5.3. Imposing an *a priori* coarse space size. We showed in the previous section that we can effectively control the condition number κ of the method by building the coarse space using two parameters α and β as presented in Theorem 10. However, this can lead to an impractically large coarse space and we now consider the context where the size n_v of the local subspace in each subdomain is chosen *a priori*. Instead of choosing the coarse space by comparing the eigenvalues to a threshold, we thus keep the eigenvectors associated with the n_v smallest eigenvalues. Once the coarse space is computed, we know what threshold would have led us to keep the same number of vectors and we can get, *a posteriori*, a bound on the condition number of the method: if λ_{n_v+1} is the lowest eigenvalue corresponding to a vector not in the coarse space, Theorem 10 ensures that

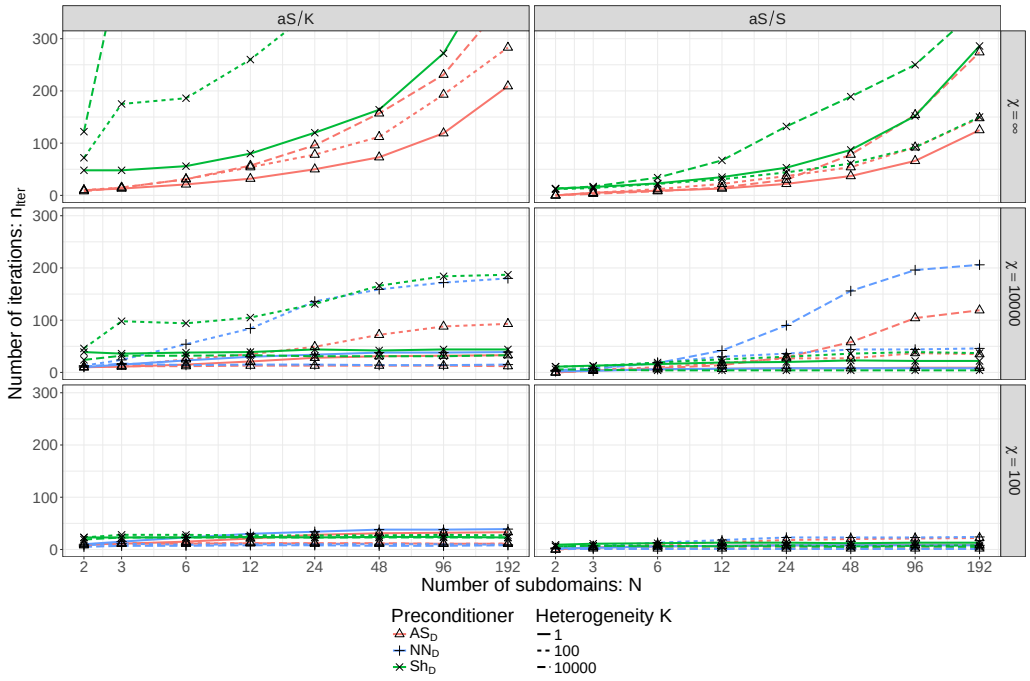

FIG. 2. *Number of iterations when imposing an* a priori *bound* χ *on the condition number.*

$\kappa(\mathcal{M}_{Sh,D}\,\mathcal{A}) \leq N_c(1+1/\lambda_{n_v+1})/\lambda_{n_v+1}$. As in Section 5.2, this bound can be improved for NN and AS preconditioners using Corollary 11 and Theorem 12:

- $\kappa(\mathcal{M}_{NN,D}\,\mathcal{A}) \leq N_c/\lambda_{n_v+1}$;
- $\kappa(\mathcal{M}_{AS,D}\,\mathcal{A}) \leq N_c(1+1/\lambda_{n_v+1})$;
- $\kappa(\mathcal{M}_{AS,2}\,\mathcal{A}) \leq (N_c+1)\,[N_c+1+(N_c+2)/\lambda_{n_v+1}]$.

The Schur matrix $\mathcal{S}$ is smaller and better conditioned [5, 27] than the original matrix $\mathcal{K}$. Furthermore, in a 2-level domain decomposition framework, eliminating the interior unknowns significantly improves the convergence by reducing the size of the coarse space needed to take into account the physical hetoregeneity in the domain [15]. In accordance with these theoretical results, Figure 4 highlights the benefits of operating on $\mathcal{S}$ (Figure 4, right) instead of $\mathcal{K}$ (left): the condition number is consistently smaller when applying any aS method on $\mathcal{S}$ instead of $\mathcal{K}$. Without a coarse space ($n_v = 0$, top), the results are consistent with Figure 1, top ($\chi = \infty$): the condition number κ increases with the number of subdomains N. Choosing $n_v = 1$, our coarse space reduces to a classical partition-of-unity coarse space [31] and is sufficient in the homogeneous case ($K = 1$, plain lines); we notice that NN,D/$\mathcal{S}$ then reduces to classical BDD where the condition number does not depend on N but remains fairly large for large values of K. However, in the heterogeneous cases ($K = 100$ or $10,000$, dashed lines), this simpler coarse space is not enough to get a scalable method: one eigenvector per high-conductivity inclusion is needed in the coarse space to build a robust method [15]. In our case, with 5 high-conductivity layers passing through all the subdomains, $n_v = 5$ eigenvectors are enough to bound the condition number for AS/$\mathcal{S}$ and NN/$\mathcal{S}$. Using the Sh/$\mathcal{S}$ method, since two eigenproblems are solved in each subdomain, 10 vectors are needed to get

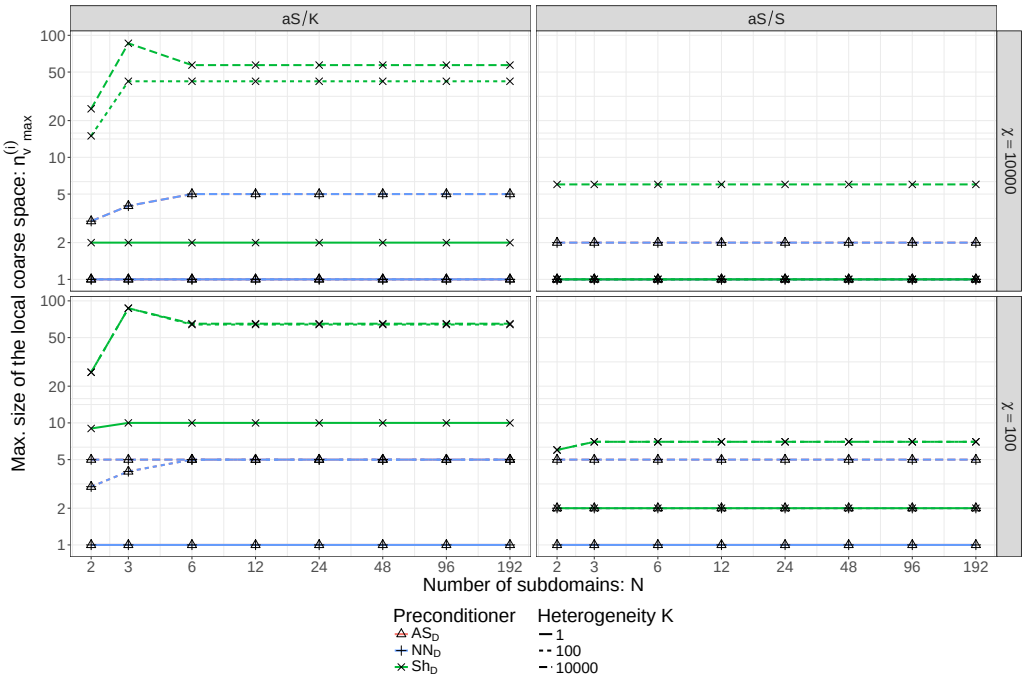

FIG. 3. *Maximum size of the local coarse space when imposing an* a priori *bound χ on the condition number. Note that AS and NN overlap with each other. In most cases, only few vectors per subdomain are enough but the least robust methods can induce a relatively large local coarse space V_i^0 in some cases.*

a good convergence (bottom right).

With a large enough coarse space, the three methods NN,D/$\mathcal{S}$, AS,2/$\mathcal{S}$ and AS,D/$\mathcal{S}$ perform quite similarly, with a slight advantage for NN. However, when the coarse space is too small ($n_v = 1$ and $K = 10,000$ for instance), AS,2/$\mathcal{S}$ and AS,D/$\mathcal{S}$ have a significantly smaller condition number than NN,D/$\mathcal{S}$, and they appear more robust. As a consequence, we will choose for our proposed high performance implementation to focus on the AS,2/$\mathcal{S}$ method (Section 5.5).

5.4. Approximate case: Empirical study of the impact of sparsification. The convergence results for approximate aS methods in sections 2, 3 and 4 apply for both aS/$\mathcal{K}$ and aS/$\mathcal{S}$ cases. However, for a matter of conciseness, we now only focus on the latter context for illustrating the impact of approximation, as the above experiments showed the numerical benefits of operating on the Schur complement. For that, we approximate the dense matrix $\mathcal{S}_i$ with a sparse matrix $\widetilde{\mathcal{S}}_i$, by dropping some entries in the matrix. This process is called *sparsification*. In a very heterogeneous medium ($K \gg 1$), some entries in $\mathcal{S}$ corresponding to couplings between unknown separated by a low-conductivity layer, are negligible. We use the symmetry-preserving strategy of dropping s_{ij} if $|s_{ij}| \leq \epsilon(s_{ii} + s_{jj})$, where ϵ is a parameter that controls the sparsity (see, e.g., [6]).

The benefits of sparsification are evaluated by assessing the proportion $nnz(LL^T)$ of non-zero elements in the Cholesky factorization $\widehat{\mathcal{S}}_i = LL^T$ of the local preconditioner. In Figure 5, we evaluate the impact of sparsification on the robustness of the method. It appears that, up to a certain level, we are still able to find a robust coarse space despite having significantly reduced

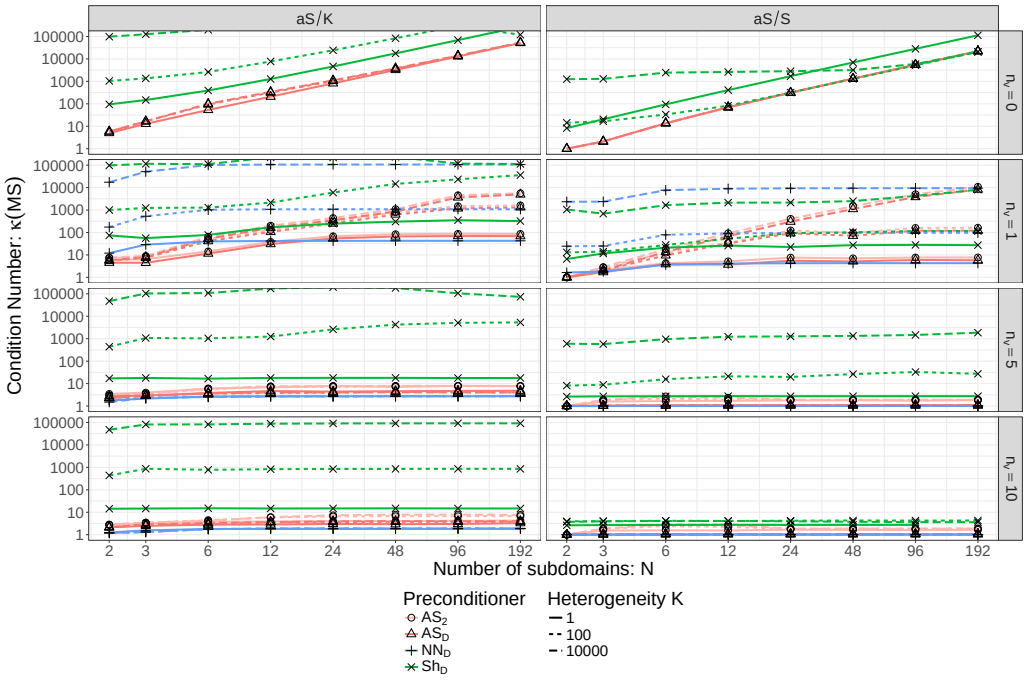

FIG. 4. *Condition number when imposing an* a priori *size n_v for the local coarse space V_i^0. We are still able to significantly reduce the condition number of the methods. The best convergence results are obtained with the AS,D/$\mathcal{S}$ method.*

the memory footprint of the preconditioner. For instance, with a sparsity parameter of $\epsilon = 0.001$, although 88.8% of the entries in the factorization of the preconditioner are dropped, our coarse space with $n_v = 5$ vectors per subdomain still significantly improves the convergence.

These results are very promising as they show we can efficiently apply an approximate scheme to reduce the complexity of two-level aS methods. However, the considered sparsification technique is delicate for ensuring an *a priori* condition number. Approximation through hierarchical matrices [17] might better fit this objective, for bounding ω_- and ω_+ and ensure theorems 10 and 12 apply. This is left for future work (see [1] for preliminary investigations in this direction) and we do not consider approximation techniques in the high performance implementation we propose below.

5.5. Performance of AS,2/$\mathcal{S}$ on a modern parallel computer. The excellent numerical properties exhibited above by the AS,2/$\mathcal{S}$ method motivated the design of an high-performance code of that variant. For that, we relied on the MaPHyS package and we added a coarse grid correction to the baseline, one-level AS/$\mathcal{S}$ variant [3] for the purpose of the present study. MaPHyS is a parallel hybrid (direct/iterative) sparse linear solver. Its *Setup* step relies on third-party sparse direct solvers for efficiently performing the elimination of the interior variables and computing the local Schur complement $\mathcal{S}_i$. Subdomains are processed concurrently, each subdomain being associated with a process. The computation of the one-level preconditioner (still within the *Setup* step) is then performed with neighbor-to-neighbor communications. The *Solve* step consists of classical preconditioned conjugate gradient iterations. In particular, global synchronizations are only re-

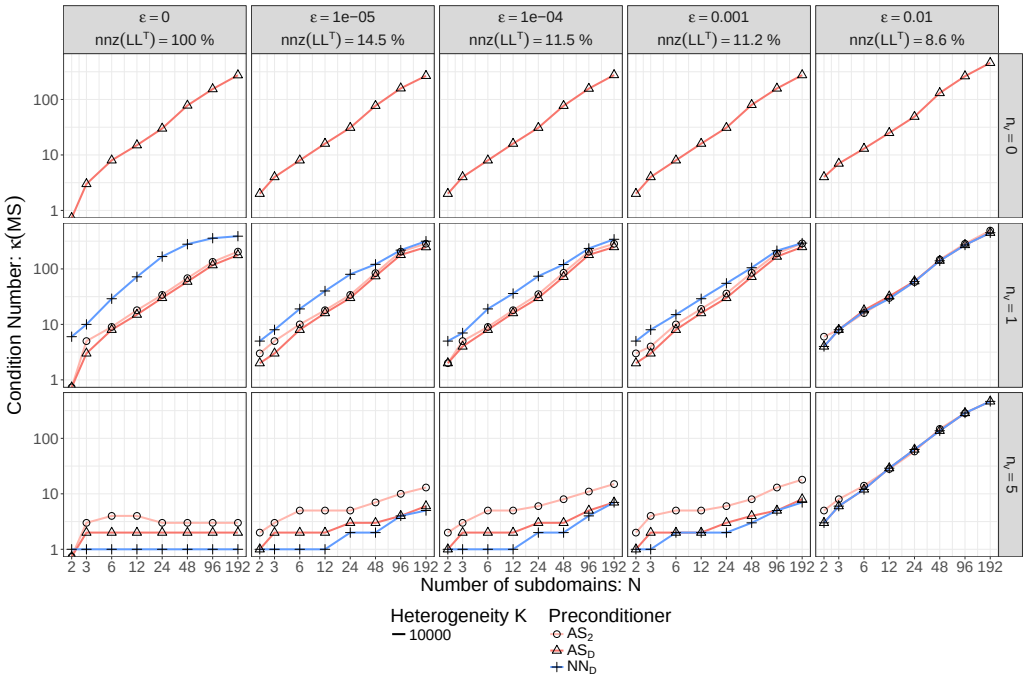

FIG. 5. *Up to a certain level, the sparsification does not break the robustness of the method: using a big enough coarse space ($n_v = 5$), it is possible to discard* 88.8% *of the entries in the factorization of the preconditioner without losing convergence.*

quired for computing dot products while the matrix-vector product can be performed concurrently on each subdomain and the application of the (one-level) preconditioner only requires neighbor-to-neighbor communications. We extended MaPHyS to ensure a coarse grid correction as follows. In the *Setup* step, the generalized eigenproblems are processed concurrently on each subdomain; the matrix associated with the resulting coarse space is then assembled and factorized using a third-party parallel sparse direct solver. In the *Solve* step, a coarse solve is added in the application of the preconditioner at each iteration. Due to the nature of the coarse space, these operations add global communications and synchronizations in the algorithm and particular care must be taken in their implementation in order to achieve good scalability and parallel efficiency. Several parallelization strategies for the coarse correction are currently investigated and will be discussed in a future work. In the current experiment, the coarse matrix $\mathcal{A}_0$ is assembled and factorized redundantly on disjoint sub-communicators (obtained by splitting the global one) in order to reduce the number of global communications during the solve step.

We now present a weak scalability study conducted on test cases similar to the ones introduced in Section 5.1, but with larger subdomains. Each subdomain is indeed a cube discretized on a $31 \times 31 \times 31$ mesh with 29,791 unknowns. There are now 6 alternating conductivity layers ($K = 10,000$), and we consider a scenario with an imposed coarse space size (as in Section 5.3) using 3 vectors per subdomain. No approximation is performed. The same stopping criterion as above is used. The experiments have been conducted on the Occigen machine at CINES. Each node is composed of two Haswell (E5-2690V3) 12-core processors running at 2.6 GHz. A subdomain is

associated with a process, binded on a CPU core. MaPHyS was compiled with Intel 17.0 and Intel MPI 2017.0.098. All dense operations are performed with the Intel Math Kernel Library (MKL) 2017 (including the Lapack `dsygvx` routine for solving the eigenproblems, that allows one to only compute a targeted subset of eigenpairs). Sparse factorizations are performed with the MUMPS 5.0.2 sparse direct solver [4] together with the ParMetis 4.0.3 partitioner [21].

Table 1 compares the behavior of our extension of MaPHyS relying on the proposed coarse grid correction described above (AS,2/$\mathcal{S}$) with the baseline, one-level version of MaPHyS [3] (AS/$\mathcal{S}$). The number of subdomains N, which is equal to the number of MPI processes and CPU cores used for the respective computation, the total number of unknowns $n = (30N+1) \times 31 \times 31$ and the size of the coarse space n_0 are provided in the table along with the maximum (among all subdomains) time in seconds needed to perform the *Setup* step, the *Solve* step or both steps (*Total*) and the number of PCG iterations performed during the *Solve* step, for both the AS/$\mathcal{S}$ method (left) and the AS,2/$\mathcal{S}$ method (right). The *Setup* step includes the time spent in the factorization of the local matrices and the computation of the local Schur complement matrix using a sequential sparse direct solver, the assembly and factorization of the local Schur complement, the solution of the generalized eigenproblems, the construction and the factorization of the coarse matrix. The *Solve* step corresponds to the PCG iterations and the final computation of the interior unknowns. We observe that the addition of the coarse correction increases the *Setup* time and the individual cost of each iteration (up to a factor 2), mainly due to the induced global communications. On the other hand, the number of iterations of AS,2/$\mathcal{S}$ remains stable, leading to a drastically overall reduced *Solve* time compared to the baseline AS/$\mathcal{S}$ method (up to a factor 37 when the 44,283,841 unknowns are distributed among 1,536 subdomains). As a consequence, in a scenario consisting of solving a linear system with a single right-hand side, the coarse grid usage reduces the total time to solution (*Setup* + *Solve*) when the number of subdomains (and CPU cores) is equal to or higher than 384. In another common application scenario where multiple (say, p), successive, right-hand sides must be solved, the total time to solution (*Setup* + p *Solve*) may then essentially be governed by the *Solve* step if p is large. In that latter case, the benefits of the coarse grid may then thus be tremendous on large scale computers.

TABLE 1

A weak scalability study was performed using the MaPHyS parallel solver. The Setup, Solve and Total times are the max among all subdomains, in seconds (s). Each subdomain is associated with one MPI process binded on one CPU core. N is the number of subdomains, n is the size of $\mathcal{K}$ and n_0 is the size of the coarse space. Without coarse correction, the Setup *time remains stable, whereas the* Solve *time grows linearly with the number of domains. The coarse correction adds to the* Setup *time but keeps the number of iterations constant, thus improving the scalability. Without coarse correction, no convergence was achieved on 3,072 domains.*

			AS/$\mathcal{S}$				AS,2/$\mathcal{S}$			
N	n	n_0	*Setup*	*Solve*	*Total*	# iter	*Setup*	*Solve*	*Total*	# iter
24	692k	72	3,64	0,47	4,12	*33*	6,13	0,30	6,44	*15*
48	1.4M	144	3,67	0,87	4,54	*62*	6,52	0,30	6,83	*15*
96	2.8M	288	3,79	1,62	5,41	*119*	6,52	0,31	6,84	*15*
192	5.6M	576	3,75	3,17	6,92	*233*	6,59	0,33	6,92	*15*
384	11.1M	1.1k	3,87	5,02	8,90	*371*	6,61	0,32	6,93	*14*
768	22.1M	2.3k	3,78	8,30	12,1	*609*	6,61	0,33	6,95	*14*
1536	44.3M	4.6k	4,13	15,1	19,2	*1,077*	6,96	0,40	7,38	*14*
3072	88.6M	9.2k	-	-	-	-	7,24	0,42	7,70	*14*

6. Conclusion. In this paper, we have proposed a new class of aS preconditioners, so-called approximate aS preconditioners. These preconditioners are fully algebraic in the sense that they do not require any other information apart from SPSD subdomain matrices. This class is wide as it consists of all aS preconditioners, provided that their coarse space results from the assembly of local components that contain the kernel of some local operators (Definition 1). In particular, it generalizes the class of approximate NN preconditioners introduced in [25] (named *generalized NN* in the original paper). We exhibited a bound on the condition number of all approximate deflated aS preconditioners (Theorem 2). This bound depends on generalized Rayleigh quotients and generalizes the result from [25] beyond the class of approximate NN methods. Applying a GenEO procedure on two well chosen generalized eigenproblems, we proposed to explicitly compute the coarse space while bounding these Rayleigh quotients leading to a bound on the condition number (hence on the number of iterations of PCG) independent of the number of subdomains. We also showed that a similar bound can be obtained when the coarse space is applied additively for the subclass of newly introduced approximate AS methods.

The results presented in this paper can be readily derived to retrieve the bounds previously obtained via generalized eigenproblems in the particular cases of AS/$\mathcal{K}$ [11, 33], NN/$\mathcal{S}$ [34] and optimized Robin (SORAS) [18]. It also generalizes these results when used with approximate local solvers. Furthermore, they allowed us to define a coarse space for the AS method applied on the Schur complement (AS/$\mathcal{S}$) [6], leading to an extremely robust substructuring method, for which the coarse space can be applied either with deflation or additively, and with the freedom of relying on an approximate local Schur complement. Numerical experiments illustrated these statements. In particular, they motivated an high-performance design of a coarse grid correction for AS/$\mathcal{S}$. We implemented it within the MaPHyS package. Parallel experiments showed the significant benefits that the resulting AS,2/$\mathcal{S}$ solver could bring.

A challenge opened by the present study is to determine an explicit procedure to perform the approximation while achieving a given *a priori* bound on the condition number. We also plan to study the effects of the method on the spectrum and on the empirical convergence of non symmetric test cases.

7. Acknowledgments. The authors would like to thank Nicole Spillane for proofreading an early draft of this manuscript. Experiments presented in sections 5.2 to 5.4 were carried out using the PLAFRIM experimental testbed, being developed under the Inria PlaFRIM development action with support from Bordeaux INP, LABRI and IMB and other entities: Conseil Régional d'Aquitaine, Université de Bordeaux and CNRS (and ANR in accordance to the programme d'investissements d'Avenir). Experiments presented in Section 5.5 were performed on the GENCI Occigen cluster at CINES by Matthieu Kuhn and Gilles Marait. We would also like to thank the anonymous referees whose constructive comments enabled us to significantly improve the manuscript. Finally, this work was partially supported by the French research agency ANR in the framework of the DEDALES project (ANR-14-CE23-0005), in particular the PhD thesis of the third author (in the alphabetical order) was funded by this project.

References.

[1] E. Agullo, E. Darve, L. Giraud, and Y. Harness, *Low-Rank Factorizations in Data Sparse Hierarchical Algorithms for Preconditioning Symmetric Positive Definite Matrices*, SIAM Journal on Matrix Analysis and Applications, 39 (2018), pp. 1701–1725, https://hal.inria.fr/hal-01940053.

[2] E. Agullo, L. Giraud, A. Guermouche, and J. Roman, *Parallel hierarchical hy-*

brid linear solvers for emerging computing platforms, Comptes Rendus Mécanique, 339 (2011), pp. 96–103, https://doi.org/10.1016/j.crme.2010.11.005, http://www.sciencedirect.com/science/article/pii/S1631072110002068 (accessed 2018-03-08).

[3] E. Agullo, L. Giraud, S. Nakov, and J. Roman, *Hierarchical hybrid sparse linear solver for multicore platforms*, report, INRIA Bordeaux, Oct. 2016, https://hal.inria.fr/hal-01379227/document (accessed 2018-03-26).

[4] P. R. Amestoy, I. S. Duff, J.-Y. L'Excellent, and J. Koster, *A fully asynchronous multifrontal solver using distributed dynamic scheduling*, SIAM Journal on Matrix Analysis and Applications, 23 (2001), pp. 15–41.

[5] S. C. Brenner, *The condition number of the Schur complement in domain decomposition*, Numerische Mathematik, 83 (1999), pp. 187–203.

[6] L. M. Carvalho, L. Giraud, and G. Meurant, *Local preconditioners for two-level non-overlapping domain decomposition methods*, Numerical linear algebra with applications, 8 (2001), pp. 207–227.

[7] B. Cockburn, J. Gopalakrishnan, and R. Lazarov, *Unified hybridization of discontinuous Galerkin, mixed, and continuous Galerkin methods for second order elliptic problems*, SIAM Journal on Numerical Analysis, 47 (2009), pp. 1319–1365.

[8] Y.-H. De Roeck and P. Le Tallec, *Analysis and test of a local domain decomposition preconditioner*, in Fourth International Symposium on Domain Decomposition Methods for Partial Differential Equations, vol. 4, 1991.

[9] C. R. Dohrmann, *A preconditioner for substructuring based on constrained energy minimization*, SIAM Journal on Scientific Computing, 25 (2003), pp. 246–258.

[10] V. Dolean, P. Jolivet, and F. Nataf, *An Introduction to Domain Decomposition Methods: Algorithms, Theory, and Parallel Implementation*, vol. 144, SIAM, 2015.

[11] Y. Efendiev, J. Galvis, R. Lazarov, and J. Willems, *Robust domain decomposition preconditioners for abstract symmetric positive definite bilinear forms*, ESAIM: Mathematical Modelling and Numerical Analysis, 46 (2012), pp. 1175–1199, http://www.esaim-m2an.org/articles/m2an/abs/2012/05/m2an110073/m2an110073.html (accessed 2017-06-08).

[12] C. Farhat, M. Lesoinne, P. LeTallec, K. Pierson, and D. Rixen, *FETI-DP: A dual–primal unified FETI method—part I: A faster alternative to the two-level FETI method*, International journal for numerical methods in engineering, 50 (2001), pp. 1523–1544.

[13] C. Farhat and F.-X. Roux, *A method of finite element tearing and interconnecting and its parallel solution algorithm*, International Journal for Numerical Methods in Engineering, 32 (1991), pp. 1205–1227.

[14] V. Frayssé and L. Giraud, *A set of conjugate gradient routines for real and complex arithmetics*, CERFACS Technical Report TR/PA/00/47, (2000).

[15] J. Galvis and Y. Efendiev, *Domain Decomposition Preconditioners for Multiscale Flows in High-Contrast Media*, Multiscale Modeling & Simulation, 8 (2010), pp. 1461–1483, https://doi.org/10.1137/090751190, http://epubs.siam.org/doi/abs/10.1137/090751190 (accessed 2016-09-15).

[16] M. J. Gander, *Optimized schwarz methods*, SIAM Journal on Numerical Analysis, 44 (2006), pp. 699–731.

[17] L. Grasedyck and W. Hackbusch, *Construction and Arithmetics of H-Matrices*, Computing, 70 (2003), pp. 295–334, https://doi.org/10.1007/s00607-003-0019-1, https://link.springer.com/article/10.1007/s00607-003-0019-1 (accessed 2018-03-26).

[18] R. Haferssas, P. Jolivet, and F. Nataf, *An Additive Schwarz Method Type Theory for*

Lions's Algorithm and a Symmetrized Optimized Restricted Additive Schwarz Method, SIAM Journal on Scientific Computing, 39 (2017), pp. A1345–A1365.

[19] P. Hénon, P. Ramet, and J. Roman, *PASTIX: A high-performance parallel direct solver for sparse symmetric positive definite systems*, Parallel Computing, 28 (2002), pp. 301–321.

[20] P. Jolivet, F. Hecht, F. Nataf, and C. Prud'homme, *Scalable Domain Decomposition Preconditioners for Heterogeneous Elliptic Problems*, in Proceedings of the International Conference on High Performance Computing, Networking, Storage and Analysis, SC '13, New York, NY, USA, 2013, ACM, pp. 80:1–80:11, https://doi.org/10.1145/2503210.2503212, http://doi.acm.org/10.1145/2503210.2503212 (accessed 2018-03-26).

[21] G. Karypis and V. Kumar, *MeTis: Unstructured Graph Partitioning and Sparse Matrix Ordering System, Version 4.0*, 2009, http://www.cs.umn.edu/~metis.

[22] A. Klawonn, M. Kuhn, and O. Rheinbach, *Adaptive coarse spaces for FETI-DP in three dimensions*, SIAM Journal on Scientific Computing, 38 (2016), pp. A2880–A2911.

[23] A. Klawonn, M. Kühn, and O. Rheinbach, *Adaptive FETI-DP and BDDC methods with a generalized transformation of basis for heterogeneous problems*, Electronic Transactions on Numerical Analysis, 49 (2018), pp. 1–27.

[24] A. Klawonn, P. Radtke, and O. Rheinbach, *A comparison of adaptive coarse spaces for iterative substructuring in two dimensions*, Electron. Trans. Numer. Anal, 45 (2016), pp. 75–106.

[25] P. Le Tallec and M. Vidrascu, *Generalized Neumann-Neumann preconditioners for iterative substructuring*, in Domain Decomposition Methods in Sciences and Engineering, 1998.

[26] J. Mandel, *Balancing domain decomposition*, International Journal for Numerical Methods in Biomedical Engineering, 9 (1993), pp. 233–241.

[27] L. Mansfield, *On the Conjugate Gradient Solution of the Schur Complement System Obtained from Domain Decomposition*, SIAM Journal on Numerical Analysis, 27 (1990), pp. 1612–1620, https://doi.org/10.1137/0727094, http://epubs.siam.org/doi/abs/10.1137/0727094 (accessed 2015-07-16).

[28] T. P. A. Mathew, *Domain Decomposition Methods for the Numerical Solution of Partial Differential Equations*, vol. 61, Springer Science & Business Media, 2008.

[29] F. Nataf, H. Xiang, V. Dolean, and N. Spillane, *A Coarse Space Construction Based on Local Dirichlet-to-Neumann Maps*, SIAM Journal on Scientific Computing, 33 (2011), pp. 1623–1642, https://doi.org/10.1137/100796376, http://epubs.siam.org/doi/abs/10.1137/100796376 (accessed 2014-11-18).

[30] A. Quarteroni and A. Valli, *Domain Decomposition Methods for Partial Differential Equations*, Oxford University Press, 1999.

[31] M. Sarkis, *Partition of unity coarse spaces: Enhanced versions, discontinuous coefficients and applications to elasticity*, Domain decomposition methods in science and engineering, (2003), pp. 149–158.

[32] N. Spillane, *Méthodes de Décomposition de Domaine Robustes Pour Les Problèmes Symétriques Définis Positifs*, PhD Thesis, Paris 6, 2014.

[33] N. Spillane, V. Dolean, P. Hauret, F. Nataf, C. Pechstein, and R. Scheichl, *Abstract robust coarse spaces for systems of PDEs via generalized eigenproblems in the overlaps*, Numerische Mathematik, 126 (2014), pp. 741–770, https://doi.org/10.1007/s00211-013-0576-y, https://link.springer.com/article/10.1007/s00211-013-0576-y (accessed 2018-03-26).

[34] N. Spillane and D. J. Rixen, *Automatic spectral coarse spaces for robust finite element tear-*

ing and interconnecting and balanced domain decomposition algorithms, International Journal for Numerical Methods in Engineering, 95 (2013), pp. 953–990, https://doi.org/10.1002/nme.4534, https://onlinelibrary.wiley.com/doi/abs/10.1002/nme.4534.

[35] A. Toselli and O. Widlund, *Domain Decomposition Methods-Algorithms and Theory*, vol. 34, Springer Science & Business Media, 2006.